

PROTOTIPO DE UN SISTEMA DE SEMAFORIZACIÓN INTELIGENTE EN LA  
CIUDAD DE BOGOTÁ PARA MEJORAR LOS TIEMPOS DE RECORRIDO DEL  
SISTEMA TRANSMILENIO

FRANZ JOSEPH ROGEELEZ CARVAJAL

UNIVERSIDAD PILOTO DE COLOMBIA  
INGENIERIA DE SISTEMAS  
BOGOTÁ  
2012

PROTOTIPO DE UN SISTEMA DE SEMAFORIZACIÓN INTELIGENTE EN LA  
CIUDAD DE BOGOTÁ PARA MEJORAR LOS TIEMPOS DE RECORRIDO DEL  
SISTEMA TRANSMILENIO

FRANZ JOSEPH ROGELÉZ CARVAJAL  
0720287

Proyecto de grado para optar el título de Ingeniero de Sistemas

Asesor:  
Andrés Mauricio Marín  
Ingeniero de Sistemas y Computación

UNIVERSIDAD PILOTO DE COLOMBIA  
INGENIERÍA DE SISTEMAS  
BOGOTÁ  
2012

Nota de aceptación

---

---

---

---

Presidente del Jurado

---

Jurado

---

Jurado

Bogotá D.C., Noviembre de 2012

Dedicado a:

A Ermelinda Carvajal Silva, mi madre por su amor y cariño.

“Si buscas resultados distintos, no utilices el mismo metodo.”

Albert Einstein

“En lugar de envidiar la naturaleza debemos emularla.”

John Holland

AGRADECIMIENTOS

El autor expresa sus agradecimientos a:

La Facultad de Ingeniería de la Universidad Piloto de Colombia y en especial a los Ingenieros y Docentes: Ignacio Hernández, Andrés Marín, Gilberto Pedraza, Fredy Perez, Giovanni Fajardo, Ibo Cerra y Fidel Barbosa.

A los investigadores y docentes Klaus Banse, Sebastian Thrun, Peter Norvig, Ralph Germ, Arne Kesting, Martin Budden, Martin Treiber, Dagoberto Sabogal y Erik Schwartz.

Sindy Mera, Carlos Rios, Wilber Palomino por su apoyo y compañía en este proyecto.

## CONTENIDO

	Pag.
INTRODUCCIÓN	10
1. PROBLEMA	11
1.1. TEMA	11
1.2. DEFINICIÓN DEL PROBLEMA	11
1.3. JUSTIFICACIÓN	12
1.4. ALCANCE	13
1.5. LIMITES	14
1.6. OBJETIVOS	14
1.6.1. GENERAL	14
1.6.2. ESPECÍFICOS	14
2. MARCO TEÓRICO	15
2.1. HISTORIA Y NOCIONES BÁSICAS	15
2.1.1. Historia1 Del Sistema De Semaforización De Bogotá	16
2.1.2. Nociones Básicas	17
2.2. HIPÓTESIS	20
2.3. VARIABLES	20
2.3.1. Variables Independientes	20
2.3.2. Variables Dependientes	20
2.3.3. Variables Intervinientes	20
3. DISEÑO METODOLÓGICO DE LA INVESTIGACIÓN	21
3.1. METODOLOGÍA PARA LA CONSTRUCCIÓN DEL PROTOTIPO	21
3.2. UNIDAD DE ANÁLISIS	22
3.3. TÉCNICAS DE RECOLECCIÓN DE LA INFORMACIÓN	22
3.4. ANÁLISIS DE LA INFORMACIÓN	22
3.5. DEFINICIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE	23
3.5.1. Plan De Desarrollo De Software	23
3.5.2. Herramientas de Desarrollo	24
3.5.3. Plan De Pruebas	24
3.5.4. Clases De Pruebas	25
4. PLAN ADMINISTRATIVO DE LA INVESTIGACIÓN	27
4.1. RECURSOS HUMANOS	27
4.2. RECURSOS FÍSICOS	28
4.3. CRONOGRAMA ACTIVIDADES	29
5. DESARROLLO METODOLOGICO	30
5.1. ANALISIS	30
5.1.1. Descripción Funcional Del Sistema	30
5.1.2. Inventario de requerimientos y casos de uso	32

5.1.3. Especificación de requerimientos y casos de uso	34
5.1.4. Diagrama de Actividades	66
5.1.5. Diagrama Conceptual	67
5.2.DISEÑO	67
5.2.1. Diagrama Entidad Relacion	67
5.2.2. Diseño de la Interfaz Grafica de Usuario (GUI)	68
5.2.3. Diagrama de alto nivel (Arquitectura)	70
5.3. IMPLEMENTACION	71
5.3.1. Metodos Inteligentes	72
5.3.2. Herramientas Estadisticas	76
5.3.3. Simuladores de Trafico	78
5.3.4. Documentacion delCodigo	80
5.3.5. Distribucion de Actividades de Codificacion	80
5.3.6. Mantenimiento	80
5.4.PRUEBAS	81
6. SUPUESTOS Y DIFICULTADES	81
6.1.ASESORIA	81
6.2.DIFICULTADES AL OBTENER INFORMACION	81
6.3.DIFICULTADES TECNICAS	81
6.4.RESULTADOS	82
7. CONCLUSIONES Y TRABAJO FUTURO	82
7.1.CONCLUSIONES	82
7.2.IMPLEMENTACION EN LA VIDA REAL	82
7.3.RECOMENDACIONES	82
8. BIBLIOGRAFÍA	83
8.1.REFERENCIAS BIBLIOGRÁFICAS	83

## GLOSARIO DE TERMINOS

**Inteligencia artificial:** Es la programación de algoritmos, que simulan la inteligencia humana.

**Trafico:** De acuerdo al artículo 2° del Código Nacional de Tránsito Ley 769 del 2000 se define como: Volumen de vehículos, peatones, o productos que pasan por un punto específico durante un periodo determinado.

**Densidad de Tráfico:** Es el volumen de vehículos y articulados que dependen de la variable tiempo.

**Movilidad:** Posibilidad de desplazamiento de forma libre, y en caso de detenerse, el tiempo en que se detiene el automóvil debe ser menor al tiempo de desplazamiento.

**Trafico Actuado:** Es la condición de crear un nuevo tráfico de forma solicitada, en cruces o vías, para esto se debe esperar a que el tráfico normal se detenga, para poder empezar el trafico actuado.

**Sistema Transmilenio:** Sistema de transporte público masivo de la ciudad de Bogotá.

**Articulado:** Son los vehículos que circulan por las troncales del sistema Transmilenio, generalmente mucho más grande que los buses normales, se le dice articulado, porque tiene una articulación en la mitad de tela, que permite doblar y girar en los recorridos.

**Prototipo:** Es el molde o modelo que se realiza para la investigación y desarrollo de alguna cosa, en este caso, es un diseño de un sistema de semaforización que se realiza para implementar en él una serie de algoritmos, el prototipo se parecerá a un sistema de semaforización real en la estructura, pero actuara de manera diferente.

**Intersección:** Es el cruce que se realiza entre una calle, y un carrera.



## **RESUMEN**

Desde que las ciudades principales del mundo comenzaron a crecer exponencialmente y de una manera muy rápida, la movilidad vial se vio afectada y las soluciones implementadas en un principio no dieron abasto con el crecimiento de la población y el sistema de semaforización actual aunque funciona es poco dinámico y eficiente para cumplir su misión de mantener un sistema vial en el que se transporte cómodamente y la densidad del tráfico sea mucho menor a la actual, esto mejoraría la calidad de vida de los habitantes de la ciudad, ya que los tiempos en que los articulados del sistema Transmilenio disminuirían de ir de un punto a otro.

## INTRODUCCIÓN

Bogotá D.C., ha cambiado sus sistemas de transporte desde hace más de 10 siglos. La circulación peatonal y a caballo de la época colonial persistió mucho tiempo, Luego, junto con la aparición del ferrocarril al principio por tracción animal y luego por fuerza eléctrica, operaron líneas de tranvías que competían con buses y taxis, hasta que el 9 de abril de 1948 fueron quemados y dañados en el Bogotazo quitando las líneas de los tranvías, luego aparecieron los buses a gasolina y trolley-buses eléctricos, estos últimos fueron desarticulados ya que con cualquier falla del electricidad generaba una parálisis total. Finalmente en 1998 se inició la construcción del sistema de transporte público masivo de la ciudad de Bogotá en adelante (Transmilenio), sistema que funciona en la actualidad acogiendo casi toda la ciudad.<sup>1</sup>

El transporte urbano en la ciudad de Bogotá, ha evolucionado para mejorar la movilidad, pero el problema persiste y ahora se suma el malestar general, el deterioro de la calidad de vida de los habitantes de la ciudad, el incremento de conflictos por la falta de tolerancia y los diversos factores son mencionados en la definición del problema. Uno de los factores que afecta principalmente al Transmilenio, es el sistema de semaforización que cuenta con un funcionamiento poco dinámico que no se adapta a las necesidades de la ciudad con un volumen de vehículos y personas tan alto.

Si se configuraran los tiempos de los semáforos en tiempo real y de forma automática dependiendo de las variaciones del tráfico, se podría mejorar la movilidad del Transmilenio y así minimizar el tiempo en que los articulados del sistema van de un punto a otro, mejorando la calidad de vida de los habitantes de la ciudad y restaurando la confianza en la administración de la ciudad.

En el presente documento se desarrolla un estudio sobre el sistema de semaforización de la ciudad de Bogotá así como su funcionamiento, su historia y sus componentes. Como resultado se generara un sistema de semaforización implementando Inteligencia Artificial, que permita el control del tráfico de forma automática y en tiempo real, configurando los tiempos de los semáforos en condición aumento o disminución del tráfico simulado.

---

1. Historia del Transporte, Francisco Pardo Téllez. Arquitecto UNIANDES. Bogotá D.C., Colombia, [http://www.todoarquitectura.com/v2/noticias/one\\_news.asp?IDNews=1045](http://www.todoarquitectura.com/v2/noticias/one_news.asp?IDNews=1045)

# **1. PROBLEMA**

## **1.1. TEMA**

Sistema de semaforización inteligente (simulador) para optimizar los tiempos del Transmilenio.

## **1.2. DEFINICIÓN DEL PROBLEMA**

La movilidad vial en el sistema de transporte público masivo de Bogotá (TRANSMILENIO) se ha visto afectada por las vías en construcción, por los pobres análisis y proyecciones en la infraestructura vial, la ausencia de cultura ciudadana por parte de los conductores y peatones y un aumento exponencial en la presencia de automóviles en la ciudad de Bogotá, además del sistema de semaforización deficiente con que cuenta la ciudad que impide que los articulados cumplan los tiempos estimados y se generen los embotellamientos de personas en las estaciones del sistema Transmilenio.

El Sistema de Semaforización vial de la ciudad de Bogotá, cuenta actualmente con 1183 intersecciones que están manejadas por 973 equipos de control marca Siemens, de los cuales el 49.3% corresponde a tecnología antigua y el 50.7% con tecnología de punta. Funciona con unos ciclos, que varían los tiempos del Semáforo en Rojo y Verde en 2 periodos diarios (Mañana – Tarde) y 3 periodos semanales (Lunes-Jueves), (Viernes), (Sábado – Domingo) y de esta manera todo el año, aunque eventualmente a través de estudios del tráfico, se realizan cambios en el tiempo de Rojo y tiempo de Verde en las programaciones semanales.

Implementando Inteligencia Artificial en la central de comunicaciones para manejar los cruces de semáforos y de esta manera el algoritmo inteligente pueda controlar el flujo de vehículos de forma automática y en tiempo real de acuerdo a las variaciones de tráfico, mejorando la movilidad en los cruces. Las variables que serían indispensables para que el algoritmo opere son: densidad de tráfico, cola de espera y cantidad de articulados.

### 1.3. JUSTIFICACIÓN

En los últimos 15 años la población de la ciudad de Bogotá ha crecido exponencialmente, y este crecimiento poblacional ha incrementado la cantidad de vehículos en la ciudad, se puede detectar rápidamente que esto genera el caos vehicular que la ciudad vive a diario, por ello se ve la necesidad de mejorar el sistema poco dinámico de semaforización de la ciudad de Bogotá, para que se capaz de adaptarse al cambio y pueda controlar el tráfico de esta forma mejorar la movilidad de la ciudad.

El simulador del sistema semaforización planteado en este proyecto busca mejorar la movilidad del sistema Transmilenio, evitando el caos que causa los vehículos que obstruyen el paso en los cruces, mejorando esto se podrá observar los beneficios, no únicamente en la disminución del tiempo del trayecto de la población de un punto a otro, además de la mejora en la calidad de vida de la ciudad, la renovación de la confianza de la administración de la ciudad, y la satisfacción de los usuarios del sistema Transmilenio.

Desde el punto de vista tecnológico, se plantea la construcción de un simulador de un sistema de semaforización inteligente que en caso de que la ciudad implemente un protocolo abierto o la empresa que cuenta con la administración puede utilizar como base este simulador para ver la eficiencia de un sistema adaptable en tiempo real que pueda proporcionar a la ciudad la movilidad que necesita, se plantea utilizar a nivel estadístico los métodos necesarios para prever el tráfico futuro y así el sistema pueda determinar las acciones pertinentes para mejorar el tráfico vial. Se incursionara en la utilización de algoritmos inteligentes que serán los utilizados por el sistema para controlar el tráfico vial.

## **1.4. ALCANCE**

Se realiza un prototipo de un sistema de semaforización desde la Calle 39 hasta la Calle 63 donde se implementara en el prototipo inteligencia artificial para controlar el sistema de semaforización de manera automática y en tiempo real de acuerdo a los cambios del tráfico.

El prototipo contara con 4 módulos.

### **Módulo 1. Control de Semaforización Grafico**

El modulo mostrara gráficamente en una simulación de los automóviles y articulados desplazándose en sentido Norte – Sur y Sur – Norte por la troncal Caracas, y en sentido Oriente – Occidente y Occidente – Oriente, por los cruces (Calle 34, Calle 39, Calle 45, Calle 53, Calle 57 y Calle 63).

### **Módulo 2. Configuración Semafórica Actual.**

El modulo mostrara en cada semáforo la programación actual donde se visualice en detalle los tiempos de rojo – amarillo y verde que tienen los semáforos en cada intersección. Y el usuario podrá cambiar de ciclo cualquier semaforo.

### **Módulo 3. Log de Cambios.**

El modulo mostrara los cambios de la programación que ha hecho el programa en el transcurso del día o mes y podrá guardar el log en un archivo.

### **Módulo 4. Estadísticas del Simulador.**

El modulo permitirá visualizar en graficas los días de mayor y menor congestión y también graficas del trafico proyectado para la siguiente semana es decir las gráficas de la densidad de tráfico pasada y futura.

Con esto se desea garantizar que el tiempo de los articulados en ir de un lugar a otro sea menor a la actual, además de mejorar el tráfico evitando la congestión generada por los automóviles que se quedan en la mitad de la intersección cuando el semáforo cambia de tiempo rojo a verde.

## **1.5. LIMITES**

El proyecto solo evaluara un tramo de la troncal caracas donde se presentan mayores embotellamientos de articulados del sistema Transmilenio, y entre ellos hay una distancia corta, los puntos críticos a evaluar son Calle 39, Calle 45, Calle 53, Calle 57 y Calle 63.

Se realiza un prototipo de simulación que implementara el algoritmo inteligente, alimentándolo de variables simuladas dadas por modelos estadísticos, las variables a estudiar son; densidad de tráfico, cantidad de articulados y tiempos de semaforización.

El programa será capaz de controlar el tráfico por medio de la configuración de los tiempos de los semáforos de rojo a verde y podrá cambiar en tiempo real esta configuración para dar paso de (Sur-Norte Norte-Sur) ó (Oriente-Occidente Occidente-Oriente) según el algoritmo inteligente lo crea conveniente.

## **1.6. OBJETIVOS**

### **1.6.1. General**

- Crear un sistema de semaforización implementando Inteligencia Artificial, que permita el control de los tiempo de ciclo rojo amarillo y verde de los semáforos en condición aumento o disminución del tráfico simulado.

### **1.6.2. Específicos**

- Implementar modelos estadísticos para la obtención de variables que alimenten el sistema inteligente.
- Moldear las variables obtenidas para que los datos que genera el simulador se parezcan a la realidad y así el experimento sea lo mas aproximado posible.
- Crear un algoritmo inteligente que sea capaz de tomar decisiones de forma autómatas y en tiempo real para controlar el tráfico y así minimizar las congestiones en el sistema Transmilenio alimentado de datos que simulen un tráfico regular como el de la ciudad de Bogotá.
- Crear un método estadístico que sea capaz de generar una toma de decisiones a partir del moldeamiento de variables ingresadas históricas y así determinar el flujo vehicular proyectado a un futuro.

## **2. MARCO TEORICO**

### **2.1 HISTORIA Y NOCIONES BÁSICAS**

#### **2.1.1 Historia<sup>1</sup> del Sistema de Semaforización de Bogotá**

En la década del 60 se instala en la ciudad la denominada “Red Blanca”, a cargo del DATT (Organización administrativa del Distrito Especial de Bogotá), con semáforos electromecánicos instalados en postes de concreto.

En 1972 ETB, como entidad del Distrito, licita y celebra el contrato que permita la instalación de una red semafórica moderna, teniendo como principal característica el manejo desde una central de control. Ésta se adjudicó en Diciembre de 1973 a las firmas Siemens S.A. (equipos de control) y Ergón S.A. (amueblamiento semafórico).

En 1974 se firma el contrato N° 0114 con ETB para que ésta administre, opere y mantenga la red semafórica por un término de 11 años. Se inició con la instalación de 180 intersecciones semaforizadas y la implementación, en 1978, de la primera central de control en Paloquemao.

La segunda central se implementó en 1982 en la zona de Chicó y la tercera en la zona de Muzú en 1985, centralizando el manejo de las diferentes intersecciones que paralelamente se iban instalando en la ciudad.

En enero de 2002 el sistema conformado por tres centros de control y 847 intersecciones semaforizadas es entregado en su totalidad al Grupo de Semaforización de la Subsecretaría Técnica de la STT, conformado en julio de 2001, para que se haga cargo de manera integral de la administración, mantenimiento y expansión del control semafórico instalado.

Actualmente cuenta con 1183 intersecciones y 973 equipos de control semafórico marca siemens, de los cuales el 49.3 % corresponde a tecnología antigua, denominados (GE Y MP)<sup>2</sup>, bajo protocolo de comunicaciones BEFA 8<sup>3</sup> y el 50.7 % restante de los equipos corresponde a tecnología de punta, denominados (MR, C800V Y C800VK)<sup>4</sup>, operados bajo el protocolo de comunicaciones BEFA 15<sup>5</sup>.

---

1. La información de la historia y los contratos del sistema semafórico de Bogotá fueron suministrados por la secretaria de movilidad, la información se aloja en el documento “SISTEMA DE CONTROL SEMAFÓRICO DE BOGOTÁ D.C”, Secretaria de Movilidad de Bogotá.

Los 973 equipos de control local se comunican con los centros de control semafórico a través de un modem mediante cable telefónico (par trenzado) que es enviado por los ductos de ETB hasta el centro de control.

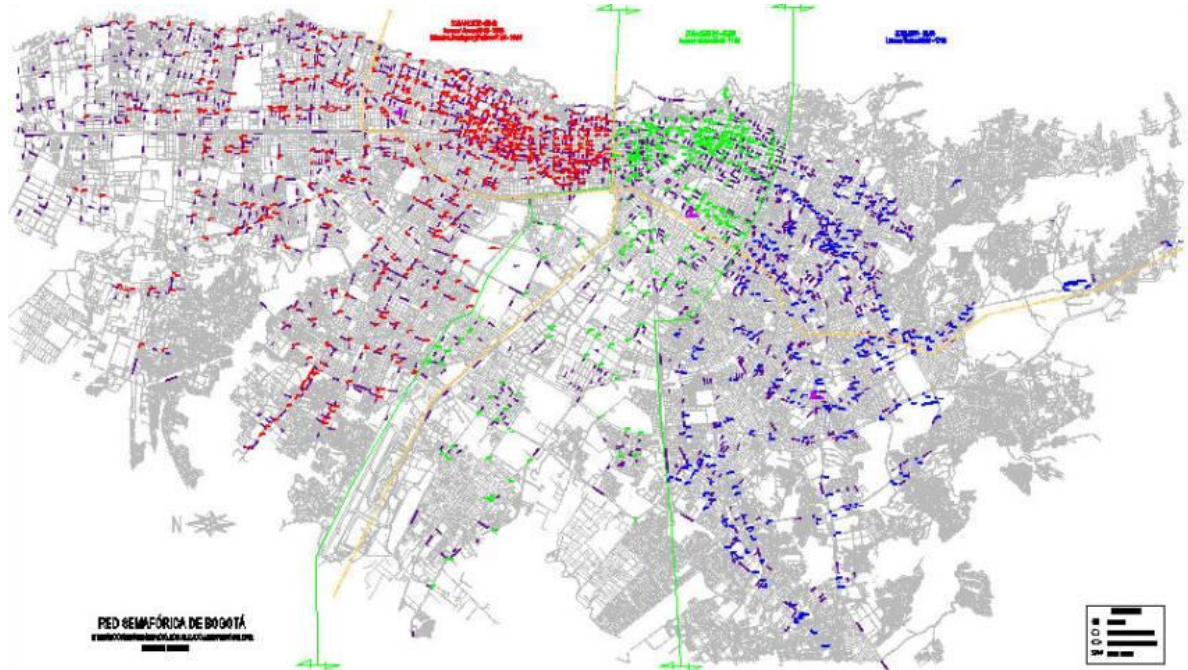


Figura 1. Referencia de la Red Semafórica de Bogotá. Fuente: Imagen de la Secretaría de Movilidad de Bogotá.

Actualmente existen 3 centros de control (PALOQUEMAO, CHICO y MUZU) que controlan la red de tráfico de la ciudad en 3 sectores que se pueden visualizar en la imagen, estos centros de control están comunicados entre ellos por fibra óptica para tener una visión total de la ciudad ya que cualquier cambio que se haga en uno de los centros puede afectar cualquier intersección en la ciudad así el centro no esté al alcance de este sector.

---

2. Hace referencia a la tecnología que usa el equipo que controla la intersección se programa con diodos y la programación debe ser manual.

3. 5. Hace referencia a los protocolos de comunicación cerrados de la empresa Siemens S.A.

4. Las tecnologías aplicadas en este formato permite a la central de control, cambiar la programación del equipo que controla la intersección, la programación se realiza en un computador que interpreta los comandos y los traduce al equipo de control.



## 2.1.2 Nociones Básicas

### 2.1.2.1 Componentes Físicos Sistema De Semaforización Electrónica

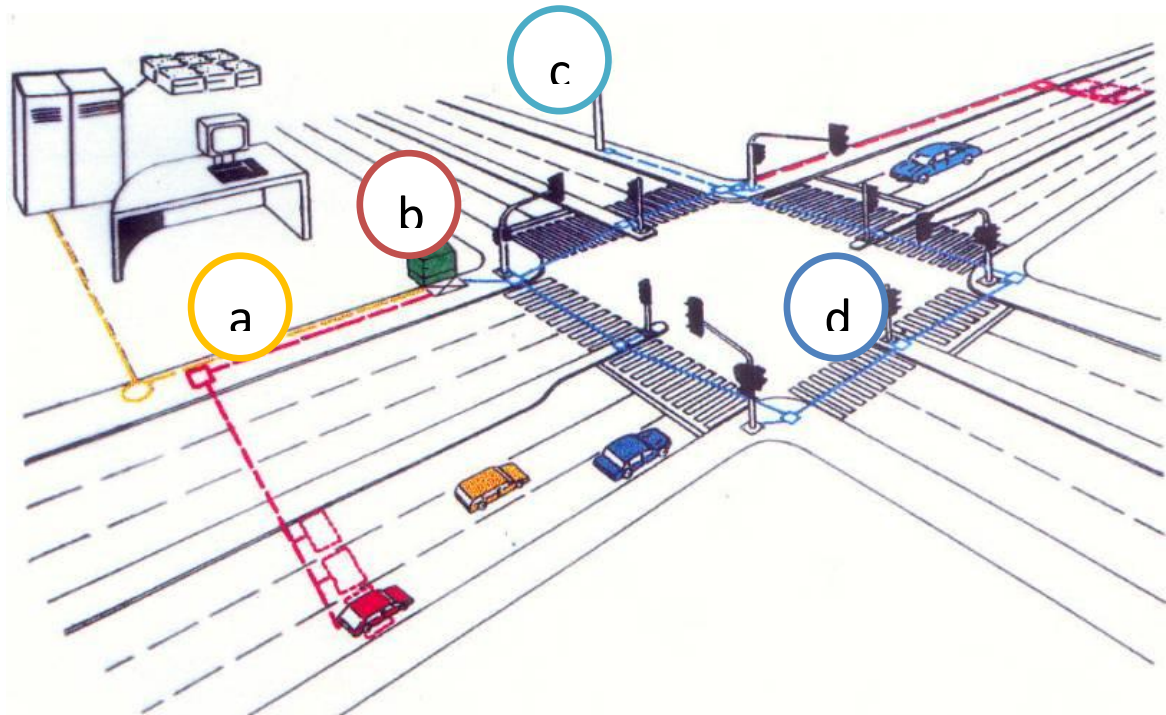


Figura 2. Dibujo de un modelo de intersección, Fuente: Imagen de la Secretaria de Movilidad de Bogotá.

#### a. Centro de Control

El centro de control cuenta con equipos siemens que realizan la conexión entre él y el equipo de control en la intersección, cuenta con los equipos y un panel del sector para realizar el monitoreo de las intersecciones, su función es verificar el funcionamiento de los semáforos y realizar las programaciones necesarias en los equipos de control.



Imagen 1. Fotografía de Centro de Control Paloquemao, Computadores que regulan los semáforos en Bogotá.

## b. Equipo de Control de la Intersección

Este equipo controla los semáforos de la intersección, la programación se realiza dependiendo del tipo tecnología, por diodos o por el controlador de código, funciona con energía suministrada por una acometida eléctrica de condensa, su función es controlar los semáforos de acuerdo a la programación que tenga o que sea enviada desde la central de control, puede existir la ocasión en que no esté conectado con la central de control por estar a mas de 5 Km en este caso la programación es de cambio manual así utilice tecnología de punta.



Imagen 2. Equipo de Control tecnología de punta, Fuente: Fotografía tomada por la Secretaria de Movilidad de Bogotá.

## c. Acometida Eléctrica

Su función es suministrar energía al equipo de control y los semáforos en caso de pérdida de energía tanto como el equipo de control y el semáforo quedaran sin abastecimiento eléctrico.

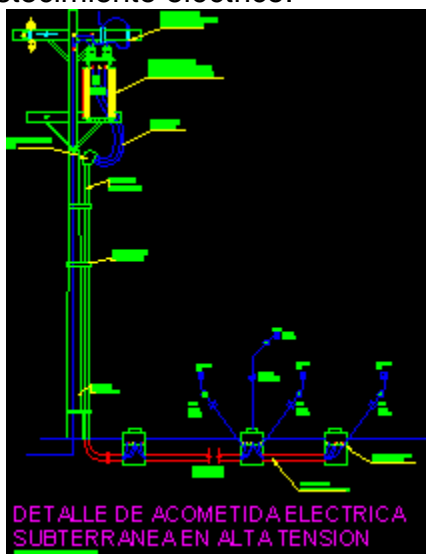


Figura 3. Acometida eléctrica subterránea en alta tensión, Fuente: Imagen tomada de la pagina web; [http://www.bibliocad.com/biblioteca/detalle-de-acometida-electrica-de-alta-tension\\_14700](http://www.bibliocad.com/biblioteca/detalle-de-acometida-electrica-de-alta-tension_14700)

#### d. Semáforos

Su función es mostrar tres colores que indicaran si el vehículo debe detenerse o debe parar, recibe energía de una acometida eléctrica y el control de este lo realiza el equipo de control.



Imagen 3. Semáforo de Bogotá,

Fuente: Imagen tomada por la secretaria de Movilidad.

#### 2.1.2.2 Red y Conexiones

Las conexiones entre los equipos de control en las intersecciones y el Centro de control se realizan por una red de interconexión telefónica adyacente a la de ETB (Empresa Telefónica de Bogotá).

La conexión se realiza por par trenzado por línea telefónica, igual a la que se utiliza para el teléfono e internet en un hogar, la conexión no puede ser superior a 5.5Km del equipo de control, a la central de control sea Chico, Muzú o Paloquemao, si supera esta distancia la intersección queda aislada de las centrales de control y el resto de las intersecciones, el equipo de control se encarga de que no hayan choques, y la programación semanal la hacen manualmente, además esto implica que el semáforo no podría enviar información de su estado actual, para saber si funciona se debe visitar periódicamente lo que implica un gasto bastante alto en cada intersección aislada.

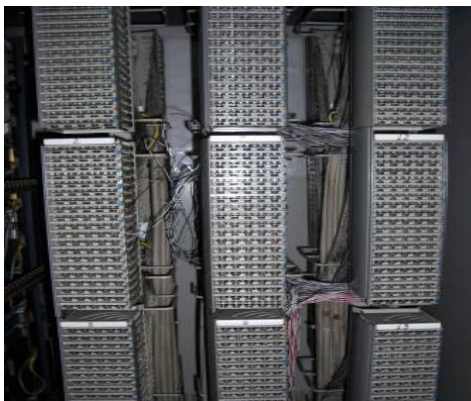


Imagen 4. Rack de Comunicación en una central de control, se puede observar los cables de par trenzado que llegan a este de los equipos de control en las intersecciones. Fuente: Imagen tomada por la Secretaria de Movilidad de Bogotá.

## **2.2 HIPÓTESIS**

El control automático de un sistema inteligente que regule los tiempos de los semáforos (ROJO/VERDE/AMARILLO) en tiempo real permitirá disminuir los tiempos de los articulados del Sistema Transmilenio en ir de un punto a otro, cumpliendo de esta manera los tiempos asignados para la ruta y así minimizar la congestión de personas en las estaciones del Sistema Transmilenio.

## **2.3 VARIABLES**

### **2.3.1 Variables Independientes**

- Tiempo del Semáforo en Rojo.
- Tiempo del Semáforo en Verde.
- Tiempo del Semáforo en Amarillo.
- Duración de la Ola verde.
- Duración del tráfico actuado.

### **2.3.2 Variables Dependientes**

- Cantidad de articulados en la vía.
- Cantidad de vehículos en el cruce.
- Velocidad promedio de los articulados.
- Velocidad de tráfico en el cruce.
- Cantidad de vehículos en cola en el cruce.
- Cantidad de artículos en cola en la vía.
- Condición actual de la vía.
- Densidad del tráfico en los vehículos.
- Densidad del tráfico en los articulados.

### **2.3.3 Variables Intervinientes**

- Longitud de vehículos
  - Vehículos = 1
  - Colectivos y Camionetas = 1.5
  - Buses Ejecutivos = 2
  - Articulados = 2.5
- Distancia entre semáforos.
- Cantidad de vías en uso.

### **3. DISEÑO METODOLOGICO DE LA INVESTIGACION**

#### **3.1 METODOLOGÍA PARA LA CONSTRUCCIÓN DEL PROTOTIPO**

Para realizar la construcción del modelo de semaforización, se va a utilizar la metodología de desarrollo iterativo (RUP) acompañado de la metodología ágil, su aplicación dentro del proyecto va a estar dada por los siguientes pasos:

Se investigan los posibles elementos que hacen parte de un prototipo de un sistema de semaforización vial, esto se hace en base a los argumentos obtenidos del marco teórico y conceptual.

Para esto es necesario que la empresa encargada del sistema de semaforización vial facilite la información necesaria para realizar el prototipo de un sistema de semaforización vial, en este caso se pide la ayuda de la Secretaria de Movilidad de Bogotá.

En base a lo consultado se evalúan los requerimientos (funcionales/no funcionales) necesarios, definiendo su contenido.

Se diseña el diagrama UML, donde se especifican las clases que interactúan con el simulador, los atributos de cada clase, y por último los métodos que deben ser acordes con los requerimientos funcionales. Se definen casos de uso, la interacción de cada caso de uso con el usuario y se realiza la especificación del funcionamiento del prototipo de semaforización vial.

A continuación se evalúan que herramientas estadísticas son necesarias para la sustentación operativa del prototipo de semaforización vial, también se evalúa el método inteligente que actuara en el prototipo.

Se realizan los algoritmos necesarios para la generación aleatoria de variables, los necesarios para el funcionamiento operativo, y el algoritmo inteligente.

Se divide de esta manera el escenario en módulos con funciones específicas y se procede a su implementación. Luego se realiza el diseño de las interfaces que harán parte del prototipo, donde contendrán las funciones de los módulos especificados anteriormente. Se procede a interconectar los módulos para crear una aplicación a partir de los componentes implementados.

Se realizan pruebas de unidad y de integración, para verificar el funcionamiento correcto de cada módulo y del simulador en su totalidad, se prueba el algoritmo inteligente con la empresa SIT Colombia, y por último con el experto de transporte para ver su viabilidad dentro del sistema de transporte de la ciudad.

### **3.2 UNIDAD DE ANÁLISIS**

En la unidad de análisis tenemos los diferentes cruces que son: Calle 34, Calle 39, Calle 45, Calle 53, Calle 57 y Calle 63 en la troncal de caracas, también se tendrán en cuenta las troncales que son paralelas a la troncal caracas, estos son la Carrera 7, la Carrera 13 y la Carrera 17.

En esos puntos en específico se evaluarán las siguientes variables:

- Densidad del tráfico de los articulados.
- Densidad del tráfico de automóviles.
- Tiempo de los semáforos (Rojo, Amarillo y Verde)

Como las variables son tomadas como datos al azar las relaciones entre variables se toman por medio de funciones probabilísticas.

### **3.3 TÉCNICAS DE RECOLECCIÓN DE LA INFORMACIÓN**

La técnica de recolección que se utilizara será la observación científica, se realizara un registro visual en los diferentes puntos del tramo seleccionados (Calle 34, Calle 39, Calle 45, Calle 53, Calle 57 y Calle 63) al mismo tiempo utilizando 6 personas que tendrán en cuenta el tiempo que se demora un articulado de ir de la Calle 34 hasta la Calle 63, también la cantidad de articulados que recorren este tramo, es decir la densidad del tráfico en el sistema Transmilenio, este proceso se realizara durante tres días en la semana, el Lunes, el Viernes y el Sábado de una semana normal.

Esta muestra real de la densidad del tráfico y los tiempos reales de los articulados, serán comparadas con la muestra simulada, de esta manera comprobar de que hay una mejora al implementar el algoritmo inteligente.

### **3.4 ANÁLISIS DE LA INFORMACIÓN**

Los datos que se recogerán en la muestra real, serán analizados estadísticamente y se utilizara únicamente para ser comparados con los datos simulados, con ello observar la respuesta que tiene el algoritmo con los datos reales, y poder ver la diferencia entre los tiempos de los articulados en ir de un punto a otro.

Esta información será tabulada, por servicio, paradas en el tramo, tiempos, y densidad del tráfico.

### 3.5 DEFINICIÓN DEL PROCESO DE DESARROLLO DE SOFTWARE A USAR

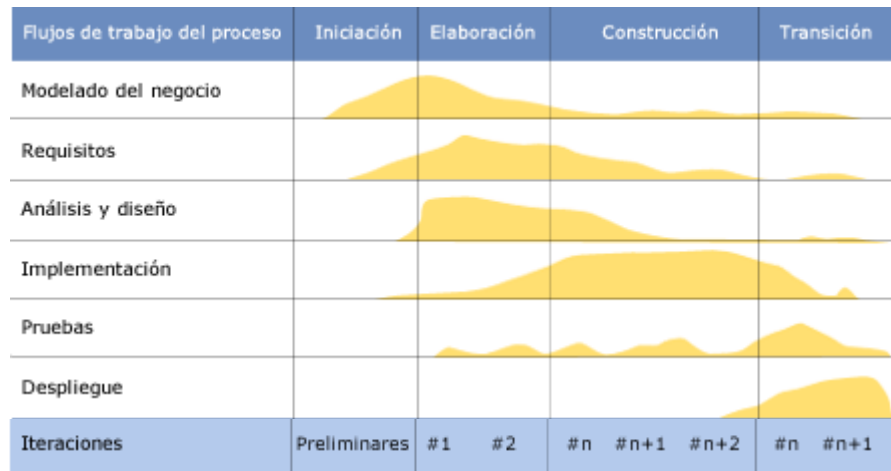


Imagen 5. Proceso de desarrollo de software RUP

Fuente Imagen: [http://es.wikipedia.org/wiki/Archivo:Rup\\_espanol.gif](http://es.wikipedia.org/wiki/Archivo:Rup_espanol.gif)

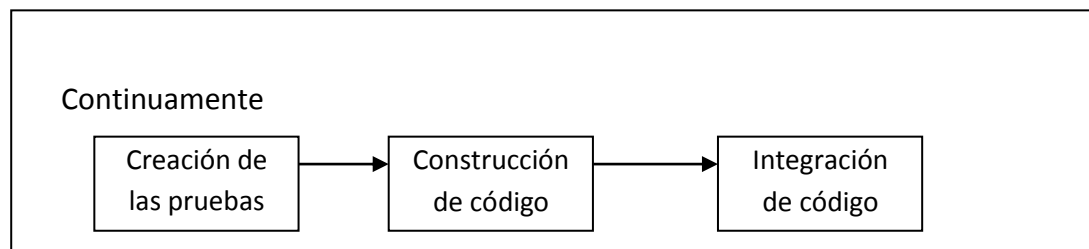


Figura 4. Parte de la metodología ágil, la parte de metodología ágil se centrara en realizar implementación continua en la fase de pruebas. Fuente: Autor.

#### 3.5.1 Plan de desarrollo de software

- **Requisitos.** En este plan se busca realizar estimaciones de recursos, costos y tiempo que puede llegar a tomar el proyecto a largo del tiempo y busca siempre clarificar los puntos más importantes dentro del desarrollo, para ajustarse a las necesidades principales del proyecto.
- **Requerimientos y análisis.** En la especificación de requerimientos se definen las necesidades identificadas por el equipo de desarrollo en la fase de construcción del modelo y análisis de la teoría, lo que permitirá que el software desempeñe el conjunto de tareas que cumpla con la idea principal del proyecto.
- **Requerimientos funcionales.** Los requerimientos funcionales están relacionados con las funciones que va a desempeñar un usuario en el sistema, estas funciones tienen un comienzo y un final y obedecen a un análisis que depende de una serie de casos de uso.

- **Requerimientos no funcionales.** Este tipo de requerimientos son transversales y no se sabe bien donde empiezan y donde terminan y no representa una acción directa en el sistema. Este tipo de requerimientos está relacionado con el ámbito del software que se desea implementar, las interfaces requeridas por el usuario y todos aquellos procesos algorítmicos obtenidos de los casos de uso.
- **Arquitectura del software.** Esta actividad define los procedimientos para cada uno de los módulos de la aplicación y también involucra el diseño y la estructura lógica de la aplicación.
- **Implementación.** La implementación es la codificación del programa y es en sí el proceso que traduce los procedimientos identificados en la fase de diseño.
- **Integración:** En la integración se comienzan a unir los diferentes módulos del proyecto de manera holística para poner en marcha toda la aplicación.

### 3.5.2 Herramientas de Desarrollo

Lenguaje de Programación: JAVA

Plataforma de programación: JAVA EE

Eclipse IDE for Java EE Developers: Eclipse Helios 3.6

Reportes y Graficas Web: BIRT Report Eclipse Versión 3.7.0 SDK

### 3.5.3 Plan de pruebas

El plan de pruebas, se realiza con el fin de establecer los parámetros, objetivos y alcances que tienen estas en el producto. Además se determinan las clases de pruebas que se deben generar, de manera ordenada y con una estrategia definida, para aplicar en el momento de la evaluación del producto, también se debe contar con unos criterios para determinar los puntos estratégicos que debe cumplir la aplicación.

**a. Información general.** En la información general se explica e indica cuál es el objetivo y el alcance que tiene el plan de pruebas en la fase final de desarrollo de la aplicación.

- El objetivo del plan administrativo de pruebas, es evaluar los requerimientos funcionales, casos de uso y tecnología planteada para el producto, con la aplicación en funcionamiento.
- El alcance del plan administrativo de pruebas, es definir los tiempos de respuesta de la aplicación.



**b. Criterios de evaluación.** Se tiene un grupo de cuatro características específicas, que cumplirá la aplicación una vez esta sea evaluada por medio de las pruebas, estas características o criterios son:

- **Funcionalidad.** Permitirá calificar si el producto maneja en forma adecuada el conjunto de funciones que satisfagan las necesidades para las cuales fue creado, en ese orden de ideas se establecen los siguientes atributos: Adecuación (evalúa si el producto cuenta con las funciones apropiadas para efectuar las tareas especificadas en la definición) y Exactitud (permite evaluar si los resultados son acordes a las necesidades para las cuales fue creado).
- **Confiabilidad.** Se refiere a la capacidad que tiene el producto para mantener su nivel de ejecución bajo condiciones normales, en un período de tiempo establecido. Las características que este estándar sugiere son: Nivel de madurez (permite medir la frecuencia de falla por errores en la aplicación) y Tolerancia a fallos (debe mantener un nivel específico de funcionamiento en caso de fallas de la aplicación).
- **Usabilidad.** Permite evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el producto, con características tales como: aprendizaje, comprensión, operatividad e interactividad.
- **Eficiencia.** Permite evaluar la relación entre el nivel de funcionamiento de la aplicación y la cantidad de recursos utilizados, el aspecto a evaluar es el comportamiento en el tiempo.

#### **3.5.4 Clases de pruebas**

Establecido el objetivo, alcance y criterios de evaluación, se han determinado las pautas para definir las pruebas que le serán realizadas al producto para valorar su calidad y funcionamiento. Las pruebas son:

**3.5.4.1 Pruebas unitarias.** Son aquellas que se realizan a las clases y métodos que componen los módulos del producto, se registra:

- Precondición para el caso de prueba
- Nombre de la clase
- Nombre del método
- Caso de prueba
- Resultado
- Objetivo de la prueba
- Criterio de la prueba
- Proceso
- Responsable
- Prueba realizada

**3.5.4.2 Pruebas de integración.** Son las pruebas de implementación, para verificar la funcionalidad del producto, según lo establecido en los requerimientos funcionales, no funcionales y casos de uso:

- Identificador de la prueba
- Estado de la prueba
- Tipo de prueba
- Objetivo de la prueba
- Ejecutado por
- Verificado por
- Flujo de eventos
- Comentarios
- Criterios de aceptación

**3.5.4.3 Pruebas de sistema.** Se realiza al momento de correr la aplicación, con el fin de tomar los tiempos de respuesta y evaluar la usabilidad y funcionalidad.

- Estado de la prueba
- Tipo de prueba
- Objetivo de la prueba
- Descripción
- Entrada
- Ejecutada por
- Verificada por
- Resultados esperados
- Resultados obtenidos
- Comentarios

## **4. PLAN ADMINISTRATIVO DE LA INVESTIGACIÓN**

### **4.1. RECURSOS HUMANOS**

Los recursos humanos necesarios para la realización del prototipo de semaforización son:

**Estadista:** Es necesario una persona que cuente con los conocimientos estadísticos sobre modelos estocásticos para recibir su orientación sobre los diferentes métodos y la manera correcta de implementarlos.

**Experto en sistemas inteligentes de transporte:** Es necesario una persona que tenga conocimientos en sistemas inteligentes para dar una orientación clara sobre los estándares o metodologías de estos sistemas y así ver si es o no conveniente aplicarlas en el proyecto actual.

**Experto en movilidad vial:** Es necesario una persona que tenga conocimientos en movilidad, obligatoriamente sobre el tráfico de la ciudad de Bogotá, y que haya trabajado en algún sector que impacte directamente con la movilidad vial.

**Desarrollador:** Para la implementación son necesarios conocimientos en el lenguaje a desarrollar que en este caso es Java. El desarrollador en este prototipo de semaforización es el mismo autor de la tesis.

**Experto en Inteligencia Artificial:** Es necesario una persona que tenga conocimientos en inteligencia artificial para recibir su orientación sobre las diferentes técnicas y la manera correcta de implementarlos. Se aclara que para este proyecto se pueden utilizar tanto una técnica inteligente como todas, siempre buscando la óptima para el desarrollo de la investigación.

**Experto en Metodología de la Investigación:** Para el desarrollo de la parte metodológica del proyecto es necesario un experto en la implementación de la metodología, para asegurarse que el proyecto se lleva por el buen camino de la investigación.

## **4.2. RECURSOS FÍSICOS**

Los recursos físicos necesarios para la realización del prototipo de semaforización son:

Un computador en el cual desarrollar el software y realizar los cálculos matemáticos y estadísticos necesarios.

Una locación para las reuniones entre los tutores y el autor de la tesis.

Es necesario la idea global de cómo funciona un sistema de semaforización real, por ello es necesario conocer uno, en este caso el sistema de semaforización de la ciudad de Bogotá se utilizara como base, la Secretaria de Movilidad de la Ciudad de Bogotá proporcionó la información necesaria para conocer el funcionamiento del sistema de semaforización de la ciudad de Bogotá, esta información esta expresada en el marco teórico.

### 4.3. CRONOGRAMA ACTIVIDADES\*

Nombre	Fecha de inicio	Fecha de fin	Recursos
<b>Requerimientos</b>	02/08/2011	17/08/2011	Andrés Mauricio Marín
<b>RNF</b>	16/08/2011	20/08/2011	Andrés Mauricio Marín
<b>Casos de Uso</b>	12/09/2011	15/09/2011	Andrés Mauricio Marín
<b>Proceso de Semaforización</b>	02/09/2011	03/09/2011	
<b>Reunión Experto SIT</b>	12/09/2011	13/09/2011	Klaus Banse SIT
<b>Reunión Experto Estadística</b>	12/09/2011	13/09/2011	
<b>Definición Estadística</b>	12/09/2011	13/09/2011	Andrés Mauricio Marín
<b>UML Diagrama de Clases</b>	14/09/2011	21/09/2011	Andrés Mauricio Marín
<b>Definición Algoritmo Inteligente</b>	19/09/2011	24/09/2011	
<b>Reunión Experto Estadística</b>	29/09/2011	04/10/2011	Dagoberto Sabogal EST
<b>Métodos estadísticos, Generación de Variables</b>	27/09/2011	12/10/2011	Dagoberto Sabogal EST
<b>Reunión Expertos Movilidad</b>	15/09/2011	16/09/2011	
<b>Algoritmos</b>	16/09/2011	14/10/2011	
<b>Interfaz Vista del SW</b>	21/09/2011	19/10/2011	Andrés Mauricio Marín
<b>Pruebas de Unidad</b>	27/09/2011	30/09/2011	Andrés Mauricio Marín
<b>Hito, Unión de Módulos</b>	30/09/2011	01/10/2011	Andrés Mauricio Marín
<b>Pruebas de Integración</b>	03/10/2011	19/10/2011	Andrés Mauricio Marín
<b>Hito, Programa Total</b>	10/10/2011	11/10/2011	Andrés Mauricio Marín
<b>Pruebas de SIT</b>	10/10/2011	19/10/2011	Klaus Banse SIT
<b>Presentación Experto TRANSPORTE</b>	19/10/2011	20/10/2011	Funcionarios SM
<b>Correcciones</b>	20/10/2011	22/10/2011	Andrés Mauricio Marín
<b>Pruebas</b>	24/10/2011	26/10/2011	Andrés Mauricio Marín
<b>Presentaciones Expertos</b>	26/10/2011	29/10/2011	Andrés Mauricio Marín
<b>Correcciones Finales</b>	31/10/2011	10/11/2011	
<b>Hito Final</b>	10/11/2011	11/11/2011	Andrés Mauricio Marín

Tabla 1. Cronograma de actividades de la implementación 02/08/11 hasta 11/11/2011 Fuente: El Autor.

\*Se presentaron varios inconvenientes para cumplir el cronograma expresados en el ítem de dificultades.

## **5. DESARROLLO METODOLOGICO**

En el presente capítulo se registran todos los resultados obtenidos durante el proceso de implementación de la metodología, donde se define la arquitectura, los requerimientos, los casos de uso, el diseño del simulador, las interfaces de los módulos del simulador, el orden de implementación y las pruebas realizadas en el escenario experimental.

### **5.1 ANALISIS**

#### **5.1.1 Descripción funcional del sistema**

El sistema contara con cuatro 4 módulos. 1 Control de Semaforización Inteligente, 2 Configuración Semafórica Actual, 3 Log de Cambios, 4 Estadísticas del Simulador.

##### **5.1.1.1. Módulo 1. Control de Semaforización Inteligente**

Este módulo corresponde al control inteligente de semáforos por medio de un proceso de simulación de tráfico particular y trafico articulado Transmilenio, donde se deben tener en cuenta los siguientes aspectos funcionales:

- a) Inicializar parámetros: este método no permitirá que las intersecciones se crucen en verde al tiempo, también inicializara las variables en sus valores por defecto.
- b) Actualizar tiempos: este método actualiza los tiempos de los semáforos en uno o todos los semáforos del sistema. Inicialmente a sus valores por defecto que se pueden establecer en el siguiente modulo.
- c) Capturar densidad del tráfico: este método utilizaría los sensores en los semáforos en la vida real, en el caso del simulador solo retornara la cantidad de vehículos y articulados que haya en cada semáforo, para determinar luego si es alta, media o baja la densidad del tráfico.
- d) Definir intersecciones prioritarias: luego de capturar la densidad de tráfico en cada semáforo define que intersecciones son prioritarias es decir que tienen una densidad alta de tráfico.
- e) Calcular tiempos semáforos: luego de definir las intersecciones con mayor cantidad de articulados y vehículos realiza cálculos para determinar los tiempos de rojo-amarillo-verde que tiene que aplicar en cada semáforo de las intersecciones para poder disminuir la cantidad de articulados y vehículos es decir disminuir la densidad del tráfico de alta a baja o media.
- f) Calcular impacto: calcula que pasa si realiza esos cambios y genera un impacto con cada uno de los tiempos que el método anterior arroja.
- g) Decisión final: Al final escoge para cada semáforo el menor tiempo para despejar este semáforo y el menor impacto en el sistema que haga el cambio ejecuta el método (1) Actualizar tiempos con los nuevos tiempos de los semáforos. En este método se utilizara el algoritmo inteligente para la toma de decisiones.

h) Verificar cambio: luego de haber ejecutado todos los cambios en las intersecciones, verifica que este cambio haya disminuido la cantidad de vehículos en la intersección, es decir que haya disminuido la densidad de tráfico.

i) Aprender: al hacer el cambio guarda en la base de datos y aprende que consecuencias y efectos tiene en el sistema el cambio realizado aprende.

(1) Si disminuyó: si efectivamente el cambio fue positivo y disminuyo la densidad del tráfico de alta a baja, de media a baja o por lo menos de alta a media, entonces vuelve al método (2) Capturar la densidad del tráfico y continua de nuevo el proceso.

(2) Si no disminuyo: y ya la densidad del tráfico es muy alta, genera una ola verde en el sistema, pero si el sistema no disminuyo y tiene una densidad del tráfico medio entonces vuelve al método (2) Capturar la densidad del tráfico y continua el proceso de nuevo.

j) Ola Verde: este método genera en toda la troncal en los límites establecidos una ola verde en ambos sentidos S-N y N-S para disminuir el tráfico la ola verde consiste en comenzar un tráfico actuado en todos los semáforos y hace que todos estos estén en verde al mismo tiempo. Ejecuta el método (7) y continúa con el proceso.

#### **5.1.1.2. Módulo 2. Configuración Semafórica Actual**

Este módulo mostrara en cada semáforo la programación actual donde se visualice en detalle los tiempos de rojo – amarillo y verde que tienen los semáforos en cada intersección. Podrá asignársele manualmente en tiempo real los valores por defecto o valores que el usuario crea convenientes.

#### **5.1.1.3. Módulo 3. Log de Cambios**

Este módulo mostrara los cambios de la programación que ha hecho el sistema inteligente en el transcurso del día o mes también podrá generar un archivo con el log de cambios.

#### **5.1.1.4. Módulo 4. Estadísticas del Simulador**

Este módulo permitirá visualizar dos estadísticas una donde muestra la información sobre la densidad del tráfico en un periodo de tiempo determinado y otra donde muestre una proyección que el sistema hace sobre la futura densidad del tráfico es decir la densidad del tráfico pasada y futura.

## 5.1.2. Inventario de requerimientos y casos de uso

### 5.1.2.1. Inventario Requerimientos no Funcionales

Requerimientos no Funcionales	
Requerimientos de Nivel de Servicio	
Nombre	Descripción
Disponibilidad	Disponible 7 días a la semana 24 horas al día
Rendimiento	Tiempo de respuesta mínimo
Recuperabilidad	En caso de fallo debe recuperarse rápido
Atributos de Calidad	
Nombre	Descripción
Confiabilidad	La fiabilidad de que el sistema está disponible, tiene un buen rendimiento y cumple con su propósito.
Persistencia	Almacenamiento de la información generada para su posterior consulta
Usabilidad	El escenario debe ser visto por el usuario final de manera fácil y clara

Tabla 2. Requerimientos no funcionales Fuente: El Autor.

### 5.1.2.2 Inventario Requerimientos Funcionales

Requerimientos funcionales		
Modulo	Requerimientos	Descripción
Control de Semafo- rización Inteligente	Ejecutar aplicacion	Inicia la aplicacion por solicitud del usuario
	Controlar semaforos	Opera los semaforos cambiando los tiempos (rojo-verde-amarillo) para disminuir el trafico
	Representación grafica simulacion	Muestra graficamente tanto el mapa donde están los vehículos y su trafico por el, como lo vehículos
Configuración Se- mafórica Actual	Mostrar configuracion actual	Muestra la configuración de los tiempos (rojo-verde-amarillo) de los semáforos
	Cambiar configuracion actual	Cambiar la configuración de los tiempos actua- les (rojo-verde-amarillo) de los semáforos
	Establecer valores por defecto	Cambiar la configuración de los tiempos asig- nados por defecto (rojo-verde-amarillo) de los semáforos
Log de Cambios	Mostrar log de cambios	Mostrar en una tabla con fecha todos los cam- bios realizados en el sistema
	Exportar a archivo log de cambios	Exportar todos los cambios del sistema a una archivo
Estadísticas del Simulador	Mostrar graficas	Mostrar graficamente los datos de la densidad del trafico, pasada, presente o futura

Tabla 3. Requerimientos funcionales Fuente: El Autor.



### 5.1.2.3 Inventario Casos de Uso

Modulo	Requerimientos Funcionales	Casos de Uso
<b>Control de Semaforización Inteligente</b>	Ejecutar aplicacion	
		Iniciar aplicacion
		Inicializar parametros
		Dar acceso al log de cambios
Controlar semaforos		
		Recorrer semaforos
		Capturar tiempos actuales
		Capturar densidad trafico (Sensor)
		Definir intersecciones prioritarias
		Calcular tiempos para disminuir trafico
		Calcular impacto de nuevos tiempos
		Verificar estados anteriores
		Decision final sobre cambio
		Actualizar tiempos
		Aprender
		Verificar cambio
		Crear trafico actuado "ola verde"
Representación grafica simulacion		
		Mostrar mapa
		Mostrar transporte
		Mostrar semaforo
<b>Configuración Semaforica Actual</b>		
	Mostrar configuracion actual	
		Recorrer semaforos
		Capturar tiempos actuales
		Mostrar tiempos actuales
Cambiar configuracion actual		
		Solicitar nuevos tiempos
		Recorrer semaforos
		Actualizar tiempos
Establecer valores por defecto		
		Solicitar tiempos por defecto
		Guardar tiempos por defecto
<b>Log de Cambios</b>		
	Mostrar log de cambios	
		Cargar cambios realizados

	Mostrar cambios realizados
Exportar a archivo log de cambios	
	Cargar cambios realizados
	Exportar un log de cambios
<b>Estadísticas del Simulador</b>	
Mostrar graficas	
	Calcular densidad del trafico actual (sensor)
	Mostrar grafica densidad del trafico actual
	Cargar datos de densidad de trafico pasado (base de datos)
	Mostrar grafica densidad del trafico pasado
	Calcular densidad del trafico futuro (métodos estadísticos)
	Mostrar densidad del trafico futuro

Tabla 4. Casos de Uso Fuente: El Autor.

### 5.1.3. Especificación de requerimientos y casos de uso

Una vez determinados los requerimientos funcionales y los requerimientos no funcionales, se procede a crear los casos de uso, con la información definida, se procede a crear las planillas de cada uno de los requerimientos y los casos de uso, esto se hace con el fin de establecer algunas características esenciales que son parte de la definición que involucran identificadores, prioridad entre otros. Se crea una planilla por cada uno de los requerimientos definidos.

#### 5.1.3.1. Requerimientos no funcionales

**RNF-001 Disponibilidad.** Es el primer requerimiento no funcional, donde el sistema debe estar disponible los 365 días del año, 24 horas al día, con un margen de error del 3% que equivalen a 10 aproximadamente,(ver Tabla 5).

<b>Identificador</b>	RNF-001	<b>Nombre:</b>	Disponibilidad
<b>Tipo</b>	Indispensable	<b>Prioridad</b>	Alta
<b>Descripción</b>	Debe tener disponibilidad 7 días 24 horas a la semana.		
<b>Impacta en estructura</b>	SI	<b>Impacta en acciones</b>	Si
<b>Descripción del impacto</b>	El requerimiento impacta al software debido a que pueden haber días celebres en los que no deba haber un acceso autorizado al sistema a no más de que lo autorice el administrador de la plataforma.		

Tabla 5. RNF-001 Fuente: El Autor.

**RNF-002 Rendimiento.** El sistema debe tener un tiempo de respuesta mínimo, en promedio para las operaciones involucradas debe ser de 3 segundos ya que son operación de alto riesgo y que necesita un alto tiempo de respuesta (ver Tabla 6).

<b>Identificador</b>	RNF-002	<b>Nombre:</b>	Rendimiento
<b>Tipo</b>	Indispensable	<b>Prioridad</b>	Alta
<b>Descripción</b>	Tiempo de respuesta mínimo.		
<b>Impacta en estructura</b>	SI	<b>Impacta en acciones</b>	Si
<b>Descripción del impacto</b>	El requerimiento impacta a las operaciones involucradas ya que trabaja sobre un marco real y los cambios deben ser rápidos para ejecutar su tarea efectivamente y no impactar en la movilidad del simulador.		

Tabla 6. RNF-002 Fuente: El Autor.

**RNF-003 Recuperabilidad.** El sistema debe recuperarse eficazmente ante cualquier fallo (ver Tabla 7).

<b>Identificador</b>	RNF-003	<b>Nombre:</b>	Recuperabilidad
<b>Tipo</b>	Indispensable	<b>Prioridad</b>	Alta
<b>Descripción</b>	Recuperarse ante cualquier fallo		
<b>Impacta en estructura</b>	SI	<b>Impacta en acciones</b>	Si
<b>Descripción del impacto</b>	El requerimiento impacta al sistema ya de que no se pueda recuperarse de un fallo el sistema es insuficiente.		

Tabla 7. RNF-003 Fuente: El Autor.

**RNF-004 Confiabilidad.** El sistema debe ser fiable y cumplir con su propósito (ver Tabla 8).

<b>Identificador</b>	RNF-004	<b>Nombre:</b>	Confiabilidad
<b>Tipo</b>	Indispensable	<b>Prioridad</b>	Alta
<b>Descripción</b>	La fiabilidad de que el sistema está disponible, tiene un buen rendimiento y cumple con su propósito		
<b>Impacta en estructura</b>	SI	<b>Impacta en acciones</b>	Si
<b>Descripción del impacto</b>	El requerimiento impacta en lo usuarios que utilizan el sistema ya que si este no cuenta con disponibilidad, rendimiento y recuperabilidad este no puede ser confiable		

Tabla 8. RNF-004 Fuente: El Autor.

**RNF-005 Persistencia.** El requerimiento de persistencia, es el que mantiene la información de una generación para su posterior consulta, (ver Tabla 9).

<b>Identificador</b>	RNF-005	<b>Nombre:</b>	Persistencia
<b>Tipo</b>	Indispensable	<b>Prioridad</b>	Alta

<b>Descripción</b>	Define como los datos son guardados en la aplicación de manera permanente.		
<b>Impacta en estructura</b>	SI	<b>Impacta en acciones</b>	Si
<b>Descripción del impacto</b>	La información debe estar almacenada en un lugar para estar disponible.		

Tabla 9. RNF-005 Fuente: El Autor.

**RNF-006 Usabilidad.** El requerimiento de persistencia, es el que mantiene la información de una generación para su posterior consulta, (ver Tabla 10).

<b>Identificador</b>	RNF-006	<b>Nombre:</b>	Usabilidad
<b>Tipo</b>	Indispensable	<b>Prioridad</b>	Alta
<b>Descripción</b>	Define que elementos se van a tener en cuenta para la facilidad de uso de la aplicación.		
<b>Impacta en estructura</b>	SI	<b>Impacta en acciones</b>	Si
<b>Descripción del impacto</b>	Dependiendo de la forma en que se definan facilidades de uso se deben realizar operaciones en la aplicación que se adecuen a esa funcionalidad.		

Tabla 10. RNF-006 Fuente: El Autor.

### 5.1.3.2. Requerimientos Funcionales

**RF-001 Ejecutar Simulador** (ver Tabla 11).

<b>Identificador</b>	RF-001	<b>Prioridad</b>	Alta
<b>Descripción</b>	Inicia la aplicación por solicitud del usuario		
<b>Inicio</b>	Inicio de la aplicación, generación de interfaces y módulos.		
<b>Fin</b>	Se debe iniciar la ejecución de la simulación según el valor de las variables dadas.		
<b>RNF</b>	Usabilidad	<b>Actor</b>	Usuario

Tabla 11. RF-001 Fuente: El Autor.

**RF-002. Controlar Semáforos.** (ver Tabla 12).

<b>Identificador</b>	RF-002	<b>Prioridad</b>	Alta
<b>Descripción</b>	Controlar los tiempos de los semáforos dependiendo de las variaciones en la densidad del tráfico, aprende de sus errores y ejecuta cambios en los tiempos.		
<b>Inicio</b>	Se controlan los tiempos de los semáforos, actualizando tiempos dependiendo de las decisiones del sistema.		
<b>Fin</b>	Cambios de tiempos en los semáforos		
<b>RNF</b>	-RNF-001-RNF-002	<b>Actor</b>	Sistema

Tabla 12. RF-002 Fuente: El Autor.

**RF-003 Representacion Grafica.** (ver Tabla 13).

<b>Identificador</b>	RF-003	<b>Prioridad</b>	Alta
<b>Descripción</b>	Representacion Grafica		
<b>Inicio</b>	Es uno de los principales requerimientos, ya que es el encargado crear la simulación del mapa		

<b>Fin</b>	Los automóviles generados como variables y con independencia en el sistema		
<b>RNF</b>	RNF-004-RNF-002-RNF-005- RNF-001	<b>Actor</b>	Sistema

Tabla 13. RF-003 Fuente: El Autor.

**RF-004 Mostrar las configuraciones actuales.** (ver Tabla 14).

<b>Identificador</b>	RF-004	<b>Prioridad</b>	Alta
<b>Descripción</b>	Muestra la configuración actual de cada semáforo		
<b>Inicio</b>	Mostrar gráficamente los tiempos actuales en cada semáforo, ROJO- VERDE-AMARILLO		
<b>Fin</b>			
<b>RNF</b>	RNF-004-RNF-002-RNF-001	<b>Actor</b>	Sistema

Tabla 13. RF-004 Fuente: El Autor.

**RF-005 Configurar Semáforos.** (ver Tabla 15).

<b>Identificador</b>	RF-005	<b>Prioridad</b>	Alta
<b>Descripción</b>	Configura los tiempos de los semáforos		
<b>Inicio</b>	Cambiar la configuración de los tiempos actuales (rojo-verde-amarillo) de los semáforos		
<b>Fin</b>			
<b>RNF</b>	RNF-004-RNF-002-RNF-001	<b>Actor</b>	Sistema

Tabla 15. RF-005 Fuente: El Autor.

**RF-006 Establecer valores por defecto.** (ver Tabla 16).

<b>Identificador</b>	RF-006	<b>Prioridad</b>	Alta
<b>Descripción</b>	Coloca los tiempos por defecto de cada parámetro		
<b>Inicio</b>	Coloca los tiempos manualmente en el parámetro por defecto de los semáforos.		
<b>Fin</b>	Tiempos de semáforos por defecto cambiados		
<b>RNF</b>	RNF-004-RNF-002-RNF-005-RNF-006	<b>Actor</b>	Usuario

Tabla 16. RF-006 Fuente: El Autor.

**RF-007 Mostrar Log de Cambios.** (ver Tabla 17).

<b>Identificador</b>	RF-007	<b>Prioridad</b>	Alta
<b>Descripción</b>	Mostrar el log de cambios en pantalla		

<b>Inicio</b>	Mostrar en una tabla con fecha todos los cambios realizados en el sistema		
<b>Fin</b>	Los cambios guardados		
<b>RNF</b>	RNF-005	<b>Actor</b>	Sistema

Tabla 17. RF-007 Fuente: El Autor.

#### RF-008 Guardar en un log. (ver Tabla 18).

<b>Identificador</b>	RF-008	<b>Prioridad</b>	Alta
<b>Descripción</b>	Guarda en un archivo de texto lo cambios hechos en el sistema		
<b>Inicio</b>	Exportar todos los cambios del sistema a una archivo		
<b>Fin</b>	Archivo de texto generado		
<b>RNF</b>	RNF-004-RNF-002-RNF-005 RNF-006	<b>Actor</b>	Usuario

Tabla 18. RF-008 Fuente: El Autor.

#### RF-009 Mostrar Estadísticas en Graficas. (ver Tabla 19).

<b>Identificador</b>	RF-009	<b>Prioridad</b>	Alta
<b>Descripción</b>	Muestra por pantalla los cambios de la densidad del tráfico.		
<b>Inicio</b>	Mostrar graficamente los datos de la densidad del trafico, pasada, presente o futura		
<b>Fin</b>	Las gráficas en pantalla.		
<b>RNF</b>	RNF-004-RNF-002-RNF-005- RNF-006	<b>Actor</b>	Usuario

Tabla 19. RF-009 Fuente: El Autor.

### 5.1.3.3. Casos de Uso.

#### 5.1.3.3.1 Casos de Uso Modulo 1 Control de Semaforizacion Inteligente

##### CU-001 Iniciar aplicación. (ver Tabla 20).

ID	CU – 001	
Nombre	Iniciar aplicaciones	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario	
DESCRIPCION DEL CASO DE USO		
Inicio de la ejecución del simulador.		
ENTRADAS		SALIDAS
Parámetros iniciales		Inicio de la simulación.
PRECONDICIONES		POSTCONDICIONES
Variables instanciadas de manera adecuada		Iniciar la simulación.
DEPENDENCIA		EXTENSION
Definir variables del sistema.		Representación Grafica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del usuario para iniciar	Carga Parámetros por medio del CU - 002
2	Carga las interfaces de usuario, inicia la simulación de vehículos y semáforos	
2	Muestra las interfaces de usuario y muestra el mapa calles y carreras del tramo, muestra la simulación de vehículos y los semáforos.	
3	El sistema inteligente de semaforización inicia actividades, inicia el CU - 003	
8	Genera log en la base de datos	
CAMINOS DE EXCEPCION		
Error en la carga de parámetros, error en la muestra de las interfaces.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad, Usabilidad		

Tabla 20. CU-001 Fuente: El Autor.

**CU-002 Inicializar parámetros.** (ver Tabla 21).

ID	CU – 002	
Nombre	Inicializar parámetros	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario, Sistema	
DESCRIPCION DEL CASO DE USO		
Se cargan los parámetros iniciales que se encuentran en el archivo .config		
ENTRADAS		SALIDAS
Ruta y nombre del archivo .config		Variables y parámetros inicializados
PRECONDICIONES		POSTCONDICIONES
Las variables del archivo .config deben estar debidamente definidas.		Iniciar la simulación.
DEPENDENCIA		EXTENSION
Programa iniciado		Instancia de parámetros
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	El usuario ingresa los valores en él .config	Valida que el archivo exista
2	Si el archivo existe lee el archivo .config Si No existe genera un error controlado	
3	Valida las variables y los datos que contienen	
4	Si están bien definidas carga las variables Sí No genera un error controlado	
5	Genera log en la base de datos	
CAMINOS DE EXCEPCION		
Error al leer las variables, error al cargar las variables, error al leer el archivo .config.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad, Usabilidad		

**Tabla 21. CU-002 Fuente: El Autor.**



**CU-003. Controlar Semáforos.** (ver Tabla 22).

ID	CU – 003		
Nombre	Controlar Semáforos		
Tipo	Implementación Obligatoria		
Prioridad	Esencial		
Actores involucrados	Sistema		
DESCRIPCION DEL CASO DE USO			
Controlar los tiempos de los semáforos dependiendo de las variaciones en la densidad del tráfico.			
ENTRADAS		SALIDAS	
La densidad del tráfico en las intersecciones y los tiempos actuales de los semáforos.		Nuevos tiempos para los semáforos para disminuir la densidad en ciertas intersecciones.	
PRECONDICIONES		POSTCONDICIONES	
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.	
DEPENDENCIA		EXTENSION	
Programa iniciado, intersecciones con tiempos.		Representación Gráfica	
FLUJO NORMAL DE EVENTOS			
No	Acciones del sistema		
1	Sistema captura la densidad del tráfico en todas la intersecciones como indica CU -		
2	Sistema captura los tiempos actuales de los semáforos como indica CU -		
3	Definir intersecciones prioritarias como indica CU -		
4	Calcula los tiempos de los semáforos para disminuir el trafico como indica CU -		
5	Calcula con los tiempos actuales el impacto que tiene cada uno de estos en el sistema como indica CU -		
6	Elige el menor tiempo y el menor impacto, aplica un algoritmo inteligente para la toma de decisiones como indica CU -		
7	Actualiza los nuevos tiempos en las intersecciones afectadas como indica CU -		
8	Genera Log de los cambios como indica CU -		
CAMINOS DE EXCEPCION			
Error al iniciar digitar los parámetros iniciales.			
CRITERIOS DE ACEPTACION			
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.			
RNF			
Persistencia, Confiabilidad, Rendimiento, Disponibilidad, Usabilidad			

Tabla 22. CU-003 Fuente: El Autor.

**CU-004 Dar acceso a log de cambios.** (ver Tabla 23).

ID	CU – 004	
Nombre	Genera log	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Guardar la hora y la fecha con el cambio realizado en la base de datos		
ENTRADAS		SALIDAS
Cambio		Log de cambio guardado
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del sistema	Toma la fecha y la hora
2	Genera el string a guardar con el formato y el parámetro de entrada	
3	Guarda los parámetros en el procedimiento almacenado de logs con la fecha, la hora y el string con el formato y el cambio realizado	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 23. CU-004 Fuente: El Autor.**

**CU-005 Recorrer semáforos.** (ver Tabla 24).

ID	CU – 005	
Nombre	Actualizar tiempos	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario, Sistema	
DESCRIPCION DEL CASO DE USO		
Se actualizan los tiempos de los semáforos, se necesita el tiempo nuevo y el semáforo o semáforos a cambiar.		
ENTRADAS		SALIDAS
Los tiempos de los semáforos y los semáforos a lo que cambiar, pueden ser ingresados por el usuario o por el sistema		Nuevos tiempos para los semáforos.
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	El Sistema envía los nuevos tiempos para los semáforos	Organiza los nuevos tiempos y los semáforos para los cuales están destinados los nuevos cambios
2	Recorre las intersecciones seleccionadas para actualizar los tiempos	
3	Cambia los tiempos en los semáforos seleccionados	
4	Guardar en el log el cambio	
CAMINOS DE EXCEPCION		
Error al cargar los tiempos, Error al guardar en la base de datos		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad, Usabilidad		

**Tabla 24. CU-005 Fuente: El Autor.**

**CU-006 Calcular tiempos actuales.** (ver Tabla 25).

ID	CU – 006	
Nombre	Calcular tiempos actuales	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Muestra el mapa en la interfaz del simulador		
ENTRADAS		SALIDAS
Solicitud		Tiempos actuales de los semáforos
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del sistema	Recorre todos los semáforos del sistema
2	Captura los tiempos rojo, verde y amarillo	
3	Envía los tiempos actuales de los semáforos	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 25. CU-006 Fuente: El Autor.**

**CU-007 Capturar densidad del tráfico.** (ver Tabla 26).

ID	CU – 007	
Nombre	Capturar densidad del tráfico	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Recorre todas las intersecciones todos los semáforos y verifica la cantidad de vehículos que hay en cada semáforo.		
ENTRADAS		SALIDAS
Solicitud de captura		Los tiempos que cuentan todos los semáforos actualmente
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del sistema	Recorre las intersecciones de todo el sistema
2	Captura el tiempo de cada semáforo los rojos, verdes y amarillos	
3	Envía los tiempos antiguos	
4	Genera el log en la base de datos	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad, Usabilidad		

**Tabla 26. CU-007 Fuente: El Autor.**

**CU-008 Definir intersecciones prioritarias.** (ver Tabla 27).

ID	CU – 008	
Nombre	Definir intersecciones prioritarias	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Decide cuales intersecciones necesitan rápidamente evacuar vehículos y/o articulados		
ENTRADAS		SALIDAS
La densidad del tráfico en las intersecciones		Las intersecciones con mayor congestión es decir con mayor densidad del tráfico.
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Salidas de CU-005 Densidad del Trafico	Recorre la densidad de todas las intersecciones
2	Evalúa cuales presentan mayor congestión	
3	Guarda las que presentan mayor congestión	
4	Envía las intersecciones con lo semáforos congestionados	
5	Generar el log de la base de datos	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad, Usabilidad		

**Tabla 27. CU-008 Fuente: El Autor.**

**CU-009 Calcular tiempos semáforos temporales.** (ver Tabla 28).

ID	CU – 009	
Nombre	Calcular tiempos semáforos	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Calcula los tiempos que serían viables cambiar en las intersecciones dadas		
ENTRADAS		SALIDAS
La densidad del tráfico en las intersecciones y los tiempos actuales de los semáforos.		Nuevos tiempos para los semáforos para disminuir la densidad en ciertas intersecciones.
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Salida CU-006 Capturar Intersecciones prioritarias	Recorre los semáforos prioritarios
2	Calcula los tiempos nuevos con los que disminuiría el trafico	
3	Toma los tiempos actuales de estas intersecciones	
4	Disminuyen los tiempos en varios rangos dependiendo de la cantidad del trafico	
5	Le da prioridad al semáforo con mayor congestión	
6	Libera los semáforos adyacentes para no traspasar la congestión a otro semáforo	
7	Envía los nuevos tiempos	
8	Genera log en la base de datos	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad		

Tabla 28. CU-009 Fuente: El Autor.

**CU-010 Calcular impacto.** (ver Tabla 29).

ID	CU – 010	
Nombre	Calcular impacto	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Define los impactos para los tiempos calculados para evacuar los vehículos		
ENTRADAS		SALIDAS
Los tiempos calculados temporales		El impacto que tiene cada uno de los tiempos en el sistema
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Salidas CU-007 nuevos tiempos para los semáforos afectados	Recorre los nuevos tiempos
2	Evalúa con los nuevos tiempos el impacto a los otros semáforos del sistema	
3	Determina en niveles de aceptación de alto, medio o bajo el impacto	
4	Compara todos los tiempos y determina cual tiempo es el menor y además que su impacto sobre el sistema sea bajo	
5	Envía la decisión de menor tiempo y menor impacto sobre todos los tiempos para los semáforos prioritarios	
6	Genera log en la base de datos	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 29. CU-010 Fuente: El Autor.**



**CU-011 Verificar decisiones anteriores.** (ver Tabla 30).

ID	CU – 011	
Nombre	Verificar decisiones anteriores	
Tipo	Implementación Adicional	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Muestra el mapa en la interfaz del simulador		
ENTRADAS		SALIDAS
Decisión probable a tomar, condiciones de la toma de decisión		Comparación de una decisión parecida tomada anteriormente con el aprendizaje de si fue buena o no
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Decisión probable a tomar, condiciones de la toma de decisión	Busca en la base de datos, decisiones con condiciones iguales o parecidas
2	Si hay una decisión parecida o igual retorna si fue buena o no y que decisión fue la que se tomo	
3	Si no envía que no encontró condiciones parecidas a la actual	
4	Guarda en el log	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 30. CU-011 Fuente: El Autor.**

**CU-012 Decisión final.** (ver Tabla 31).

ID	CU – 012	
Nombre	Decisión final	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Elige el menor tiempo y el menor impacto, aplica un algoritmo inteligente para la toma de decisiones.		
ENTRADAS		SALIDAS
Tiempos temporales E impactos para estos tiempos.		Nuevos tiempos para los semáforos.
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Salidas CU-008 tiempos bajos impacto mínimo	Analiza las alternativas enviadas luego de revisar el impactos de los tiempos
2	Verifica con resultados anteriores si no tiene no realiza la verificación	
3	Hace uso del algoritmo inteligente para tomar la opciones adecuadas (Etapa de Duda)	
4	Con los nuevos tiempos ejecuta los cambios en los semáforos afectados	
5	Envía los tiempos para actualizar los semáforos	
6	Genera log de cambios en la base de datos	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 31. CU-012 Fuente: El Autor.**

**CU-013 Actualizar tiempos.** (ver Tabla 32).

ID	CU – 013	
Nombre	Actualizar tiempos	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario, Sistema	
DESCRIPCION DEL CASO DE USO		
Se actualizan los tiempos de los semáforos, se necesita el tiempo nuevo y el semáforo o semáforos a cambiar.		
ENTRADAS		SALIDAS
Los tiempos de los semáforos y los semáforos a lo que cambiar, pueden ser ingresados por el usuario o por el sistema		Nuevos tiempos para los semáforos.
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	El Sistema envía los nuevos tiempos para los semáforos	Organiza los nuevos tiempos y los semáforos para los cuales están destinados los nuevos cambios
2	Recorre las intersecciones seleccionadas para actualizar los tiempos	
3	Cambia los tiempos en los semáforos seleccionados	
4	Guardar en el log el cambio	
CAMINOS DE EXCEPCION		
Error al cargar los tiempos, Error al guardar en la base de datos		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad, Usabilidad		

**Tabla 32. CU-013 Fuente: El Autor.**

**CU-014 Aprender.** (ver Tabla 33).

ID	CU – 014	
Nombre	Aprender	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Guarda en la base de datos y aprende las consecuencias del cambio realizado. Observa si el cambio fue positivo o negativo es decir, disminuyo o aumento la densidad del tráfico, aprende del error y cuando vuelve a evaluar las opciones puede determinar resultados con opciones escogidas anteriormente.		
ENTRADAS		SALIDAS
Tiempos anteriores, Nuevos Tiempos, Densidad de tráfico anterior, Nueva densidad de trafico		Aprende
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del sistema	Recorre las intersecciones y captura los tiempos actuales y la densidad de tráfico actual
2	Captura los tiempos inmediatamente anteriores y la densidad de la base de datos	
3	Compara los tiempos actuales con los anteriores	
4	Compara la densidad de tráfico en la intersecciones prioritarias	
5	Verifica el impacto real del cambio realizado en el sistema	
6	Guarda el impacto real en la base de datos para posteriores consultas	
7	Guarda la densidad y si el cambio de tiempos fue positivo o negativo en el sistema	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 33. CU-014 Fuente: El Autor.**

**CU-015 Verificar cambio.** (ver Tabla 34).

ID	CU – 015	
Nombre	Verificar decisiones anteriores	
Tipo	Implementación Adicional	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Muestra el mapa en la interfaz del simulador		
ENTRADAS		SALIDAS
Decisión probable a tomar, condiciones de la toma de decisión		Comparación de una decisión parecida tomada anteriormente con el aprendizaje de si fue buena o no
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Decisión probable a tomar, condiciones de la toma de decisión	Busca en la base de datos, decisiones con condiciones iguales o parecidas
2	Si hay una decisión parecida o igual retorna si fue buena o no y que decisión fue la que se tomo	
3	Si no envía que no encontró condiciones parecidas a la actual	
4	Guarda en el log	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 34. CU-015 Fuente: El Autor.**

**CU-016 Crear trafico actuado (Generar Ola Verde.)** (ver Tabla 35).

ID	CU – 016	
Nombre	Generar Ola Verde	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
En caso de que el sistema genera una ola verde, es decir coloca todos los semáforos en verde para mejorar el flujo de vehículos		
ENTRADAS		SALIDAS
Intersecciones congestionadas		Ola verde realizada Guarda log del cambio en la base de datos
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del sistema	Captura densidad de trafico
2	Ubica las intersecciones aledañas que tienen mayor densidad de trafico	
3	Verifica los nuevos tiempos que tendrían todo los semáforos en ambos sentidos en verde para disminuir la congestión.	
4	Calcula el impacto que tendrían todas las opciones de tiempos sobre el sistema en tiempo verde	
5	Si el impacto es bajo o medio realiza el cambio con el menor tiempo y menor impacto	
6	Si el impacto sobre el sistema es alto no se realiza la ola verde y se continua con el proceso normal	
7	Guarda Log en la base de datos	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Persistencia, Confiabilidad, Rendimiento, Disponibilidad		

Tabla 35. CU-016 Fuente: El Autor.

**CU-017 Mostrar de mapa.** (ver Tabla 36).

ID	CU – 017	
Nombre	Muestra de mapa	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Muestra el mapa en la interfaz del simulador		
ENTRADAS		SALIDAS
Inicia con el programa		Simulación en pantalla
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del sistema	Inicia las calles y carreras en el sistema
2	Muestra las calles y las carreras en el sistema	
3	Inicia la simulación	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 36. CU-017 Fuente: El Autor.**

**CU-018 Mostrar transporte vehículos-articulados.** (ver Tabla 37).

ID	CU – 018		
Nombre	Muestra de vehículos-articulados		
Tipo	Implementación Obligatoria		
Prioridad	Esencial		
Actores involucrados	Sistema		
DESCRIPCION DEL CASO DE USO			
Muestra los vehículos, y articulados simulados.			
ENTRADAS		SALIDAS	
Inicia con el programa		Simulación en pantalla	
PRECONDICIONES		POSTCONDICIONES	
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.	
DEPENDENCIA		EXTENSION	
Programa iniciado, intersecciones con tiempos.		Representación Gráfica	
FLUJO NORMAL DE EVENTOS			
No	Acción del actor	Respuesta del sistema	
1	Solicitud del sistema	Inicia la carga de cantidad de vehículos y el horario para los vehículos y articulados	
2	Maneja cada articulado y vehículo independiente		
3	Dependiendo del horario muestra una cantidad determinada de vehículos		
4	Organiza los vehículos y los distribuye a través de la simulación y del mapa		
5	Inicia la simulación		
CAMINOS DE EXCEPCION			
Error al iniciar digitar los parámetros iniciales.			
CRITERIOS DE ACEPTACION			
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.			
RNF			
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad			

**Tabla 37. CU-018 Fuente: El Autor.**



**CU-019 Mostrar semáforo.** (ver Tabla 38).

ID	CU – 019	
Nombre	Muestra de semáforos	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Sistema	
DESCRIPCION DEL CASO DE USO		
Muestra los semáforos y coloca los tiempos dependiendo de cómo estén configurados		
ENTRADAS		SALIDAS
Inicia con el programa		Simulación en pantalla
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del sistema	Muestra los semáforos en el mapa
2	Actualiza o inicializa los tiempos en los semáforos y muestra los colores de estos, los maneja independientemente	
4	Muestra los semáforos y sus tiempos dependiendo de la configuración actual en el sistema	
5	Inicia la simulación	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 38. CU-019 Fuente: El Autor.**

### 5.1.3.3.2 Casos de Uso Modulo 2 Configuracion Semaforica Actual

**CU-020 Mostrar tiempos actuales.** (ver Tabla 39).

ID	CU – 020	
Nombre	Mostrar las configuraciones actuales	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario	
DESCRIPCION DEL CASO DE USO		
Muestra la configuración actual de cada semáforo		
ENTRADAS		SALIDAS
Solicitud del usuario		Configuración actual de todos los semáforos
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del usuario	Recorre las intersecciones
2	Captura los tiempos de todos los semáforos	
3	Muestra por pantalla los tiempos actuales de los semáforos rojo, amarillo y verde	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad		

**Tabla 39. CU-020 Fuente: El Autor.**

**CU-021 Solicitar nuevos tiempos.** (ver Tabla 40).

ID	CU – 015	
Nombre	Establecer valores por defecto	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario	
DESCRIPCION DEL CASO DE USO		
Coloca los tiempos manualmente en el parámetro por defecto de los semáforos.		
ENTRADAS		SALIDAS
Tiempos nuevos por defecto		Nuevos tiempos por defecto ingresados
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Usuario inicializa las variables por defecto en el módulo correspondiente	Evalúa la variables
2		Verifica que los tipos de las variables sea el correspondiente
3	Reingresa las variables mal digitadas	Si las variables están mal ingresadas envía un error al usuario
4		SI las variables están bien ingresadas, cambia la configuración inicial y muestra un mensaje de que el cambio se realizó satisfactoriamente
5	Guarda el log en la base de datos con el nombre del usuario que realizo el cambio	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad, Recuperabilidad, Persistencia		

**Tabla 40. CU-021 Fuente: El Autor.**

**CU-022 Establecer los tiempos actuales de los semáforos en tiempo real.**  
(ver Tabla 41).

ID	CU – 022	
Nombre	Establecer los tiempos actuales de los semáforos en tiempo real	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario	
DESCRIPCION DEL CASO DE USO		
Configurar los tiempos de los semáforos, se usa para pruebas, se cambian los tiempos de los semáforos manualmente.		
ENTRADAS		SALIDAS
Tiempos nuevos para semáforos en tiempo real		Nuevos tiempos para los semáforos
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del usuario	Sistema se dispone a recibir los nuevos tiempos mostrando los semáforos y los tiempos actuales de estos
	Usuario ingresa los nuevos tiempos para los semáforos que el quiera	Verifica las variables ingresadas corresponden a los datos
		Si están bien ingresadas actualiza los tiempos a los semáforos escogidos y envía un mensaje al usuario con el cambio se hizo satisfactoriamente
	Usuario reingresa la variables	Si no están bien ingresadas envía un mensaje de error y pide que se ingresen de nuevo
5	Guarda el log en la base de datos con el nombre del usuario que realizo el cambio	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad, Recuperabilidad, Persistencia		

Tabla 41. CU-022 Fuente: El Autor.

**CU-023 Solicitar tiempos por defecto.** (ver Tabla 42).

ID	CU – 023	
Nombre	Establecer valores por defecto	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario	
DESCRIPCION DEL CASO DE USO		
Coloca los tiempos manualmente en el parámetro por defecto de los semáforos.		
ENTRADAS		SALIDAS
Tiempos nuevos por defecto		Nuevos tiempos por defecto ingresados
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Usuario inicializa las variables por defecto en el módulo correspondiente	Evalúa la variables
2		Verifica que los tipos de las variables sea el correspondiente
3	Reingresa las variables mal digitadas	Si las variables están mal ingresadas envía un error al usuario
4		SI las variables están bien ingresadas, cambia la configuración inicial y muestra un mensaje de que el cambio se realizó satisfactoriamente
5	Guarda el log en la base de datos con el nombre del usuario que realizo el cambio	
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad, Recuperabilidad, Persistencia		

**Tabla 42. CU-023 Fuente: El Autor.**

### 5.1.3.3.3 Casos de Uso Modulo 3 Log de Cambios

**CU-024 Mostrar cambios realizados.** (ver Tabla 43).

ID	CU – 024		
Nombre	Ver Log de Cambios		
Tipo	Implementación Obligatoria		
Prioridad	Esencial		
Actores involucrados	Usuario		
DESCRIPCION DEL CASO DE USO			
Generar Log de Cambios			
ENTRADAS		SALIDAS	
Solicitud del usuario		Muestra por pantalla el log de cambios	
PRECONDICIONES		POSTCONDICIONES	
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.	
DEPENDENCIA		EXTENSION	
Programa iniciado, intersecciones con tiempos.		Representación Gráfica	
FLUJO NORMAL DE EVENTOS			
No	Acción del actor		Respuesta del sistema
1	Solicitud del usuario		Accede a la tabla y trae los cambios realizados en el día o la fecha seleccionada
			Si la fecha es correcta Muestra por pantalla los cambios que se a realizado
			Si no es correcta pide al usuario que ingrese de nuevo la fecha.
CAMINOS DE EXCEPCION			
Error al iniciar digitar los parámetros iniciales.			
CRITERIOS DE ACEPTACION			
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.			
RNF			
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad, Recuperabilidad, Persistencia			

Tabla 43. CU-024 Fuente: El Autor.

**CU-025 Exportar en un log. (ver Tabla 44).**

ID	CU – 025		
Nombre	Guardar en un log		
Tipo	Implementación Obligatoria		
Prioridad	Esencial		
Actores involucrados	Usuario		
DESCRIPCION DEL CASO DE USO			
Guarda en un archivo de texto lo cambios hechos en el sistema			
ENTRADAS		SALIDAS	
Solicitud del usuario		Archivo con los cambios del sistema	
PRECONDICIONES		POSTCONDICIONES	
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.	
DEPENDENCIA		EXTENSION	
Programa iniciado, intersecciones con tiempos.		Representación Gráfica	
FLUJO NORMAL DE EVENTOS			
No	Acción del actor		Respuesta del sistema
1	Usuario solicita el log de cambios		Carga los datos de la base de datos y pide al usuario donde y con qué nombre guardar el archivo
2	Escoge el directorio y el nombre del archivo a guardar		Guarda en el archivo los cambios que están registrados en la base de datos
CAMINOS DE EXCEPCION			
Error al iniciar digitar los parámetros iniciales.			
CRITERIOS DE ACEPTACION			
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.			
RNF			
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad, Recuperabilidad, Persistencia			

**Tabla 44. CU-025 Fuente: El Autor.**

### 5.1.3.3.1 Casos de Uso Modulo 4 Estadísticas del Simulador

**CU-026 Representar Estadísticas.** (ver Tabla 45).

ID	CU – 026	
Nombre	Representar Estadísticas	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario	
DESCRIPCION DEL CASO DE USO		
Muestra por pantalla los cambios de la densidad del tráfico.		
ENTRADAS		SALIDAS
Solicitud del usuario		Estadísticas en pantalla
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del usuario	Le muestra las dos estadísticas y los tipos de grafica a escoger
	Escoge el tipo de gráfica y la estadística futura, pasada o presente	Trae la información de la base de datos para las gráficas futura y pasada y para la ´presente captura la densidad del tráfico actual
		Muestra la estadística por pantalla elegida por el usuario y con el tipo de grafica escogido
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad, Persistencia		

Tabla 45. CU-026 Fuente: El Autor.



**CU-027 Calcular densidad del trafico futuro.** (ver Tabla 46).

ID	CU – 027	
Nombre	Calcular densidad del trafico futuro	
Tipo	Implementación Obligatoria	
Prioridad	Esencial	
Actores involucrados	Usuario	
DESCRIPCION DEL CASO DE USO		
Muestra por pantalla los cambios de la densidad del tráfico.		
ENTRADAS		SALIDAS
Solicitud del usuario		Estadísticas en pantalla
PRECONDICIONES		POSTCONDICIONES
Todas las variables del sistema deben estar debidamente definidas, la simulación iniciada.		Tiempos de las intersecciones.
DEPENDENCIA		EXTENSION
Programa iniciado, intersecciones con tiempos.		Representación Gráfica
FLUJO NORMAL DE EVENTOS		
No	Acción del actor	Respuesta del sistema
1	Solicitud del usuario	Le muestra las dos estadísticas y los tipos de grafica a escoger
	Escoge el tipo de gráfica y la estadística futura, pasada o presente	Trae la información de la base de datos para las gráficas futura y pasada y para la ´presente captura la densidad del tráfico actual
		Muestra la estadística por pantalla elegida por el usuario y con el tipo de grafica escogido
CAMINOS DE EXCEPCION		
Error al iniciar digitar los parámetros iniciales.		
CRITERIOS DE ACEPTACION		
Se deben encontrar definidas todas las variables del sistema inteligente de semaforización.		
RNF		
Usabilidad, Confiabilidad, Rendimiento, Disponibilidad, Persistencia		

**Tabla 46. CU-027 Fuente: El Autor.**

#### 5.1.4 Diagrama de Actividades

Los flujos de proceso asociados a los casos de uso principales se describieron con los siguientes diagramas de actividades, hechos con la herramienta de modelamiento de procesos Bizagi con licencia Freeware

##### 5.1.4.1 Diagrama Control Inteligente de Semaforizacion

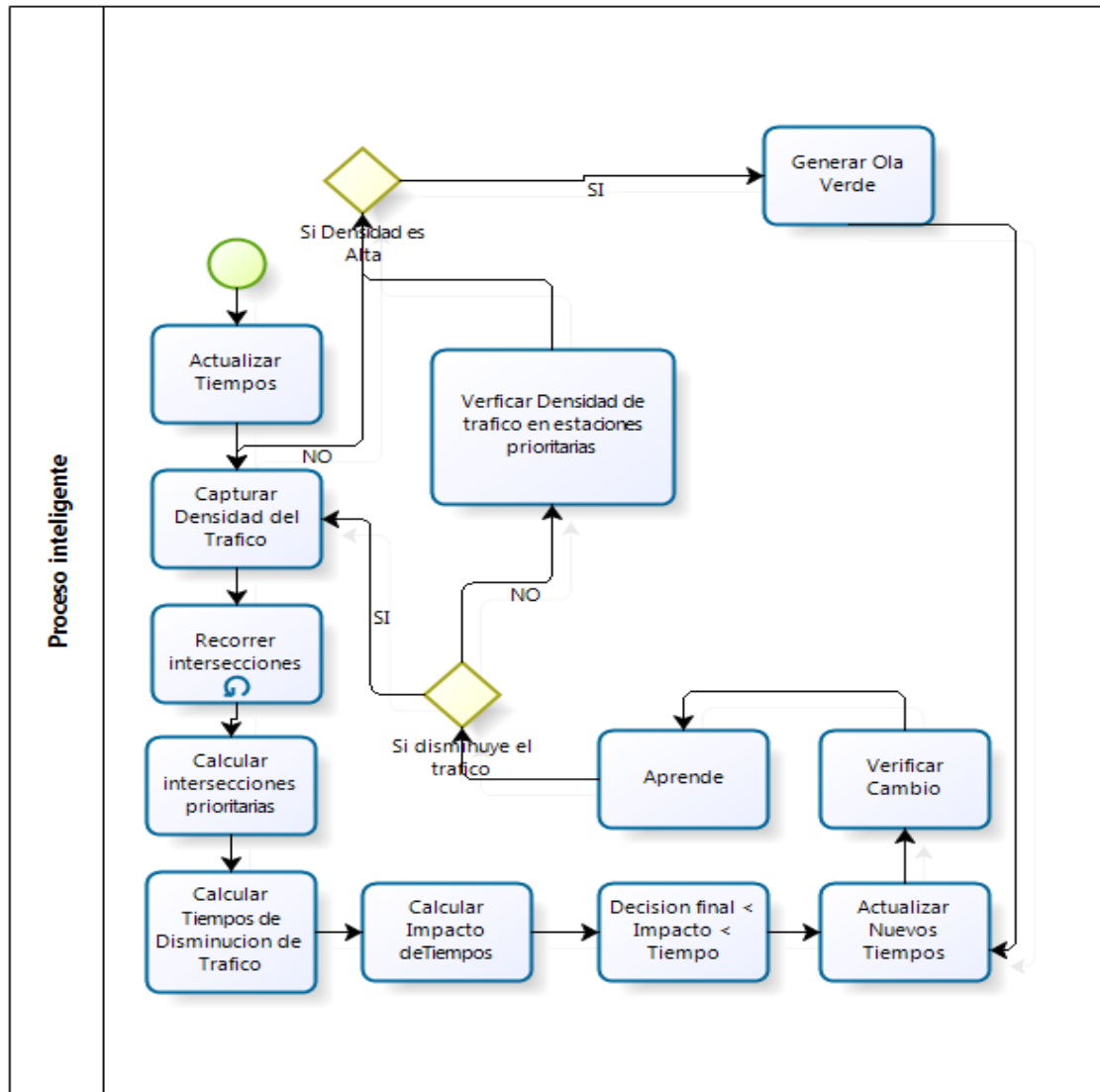


Figura 5. BPMN Proceso inteligente descrito en el caso de uso CU-003 Controlar Semaforos, Fuente: Autor.

### 5.1.5 Diagrama conceptual

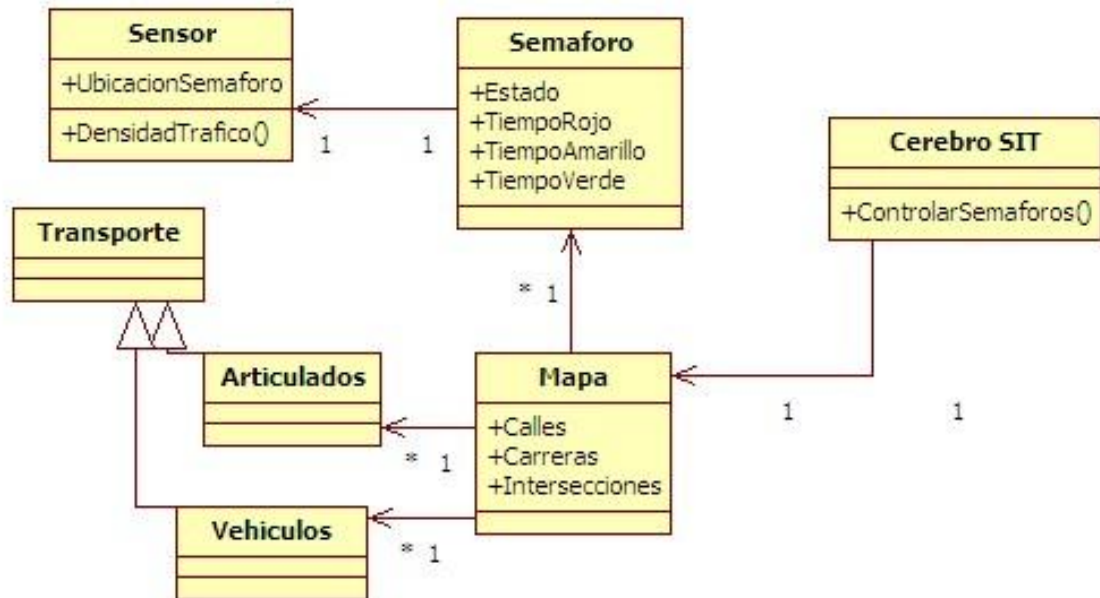


Figura 6. Diagrama conceptual

Fuente: Autor.

## 5.2 DISEÑO

### 5.2.1 Diagrama Entidad Relacion

Este diagrama se encuentra en fase de elaboración ya que se estiman el diseño de la base de datos cumpliendo los casos de uso y minizando en tiempo de búsqueda para alguna operación, con ello mejorar las búsquedas desde el diseño de este y no optimizarlas después.

Figura 7. Diagrama Entidad - Relacion

Fuente: Autor.

### 5.2.2 Diseño de la Interfaz Grafica de Usuario (GUI)

Como se detallo en la descripción funcional del sistema, este cuenta con 4 modulos, que ahora vamos a visualizar gráficamente para cumplir con el requerimiento no funcional RNF-006 Usabilidad.

#### 5.2.2.1 Modulo 1. Control de Semaforización Inteligente



Imagen 6. Modulo 1 . Control de Semaforizacion Inteligente Fuente: Autor.

### 5.2.2.2 Modulo 2. Configuración Semaforico Actual

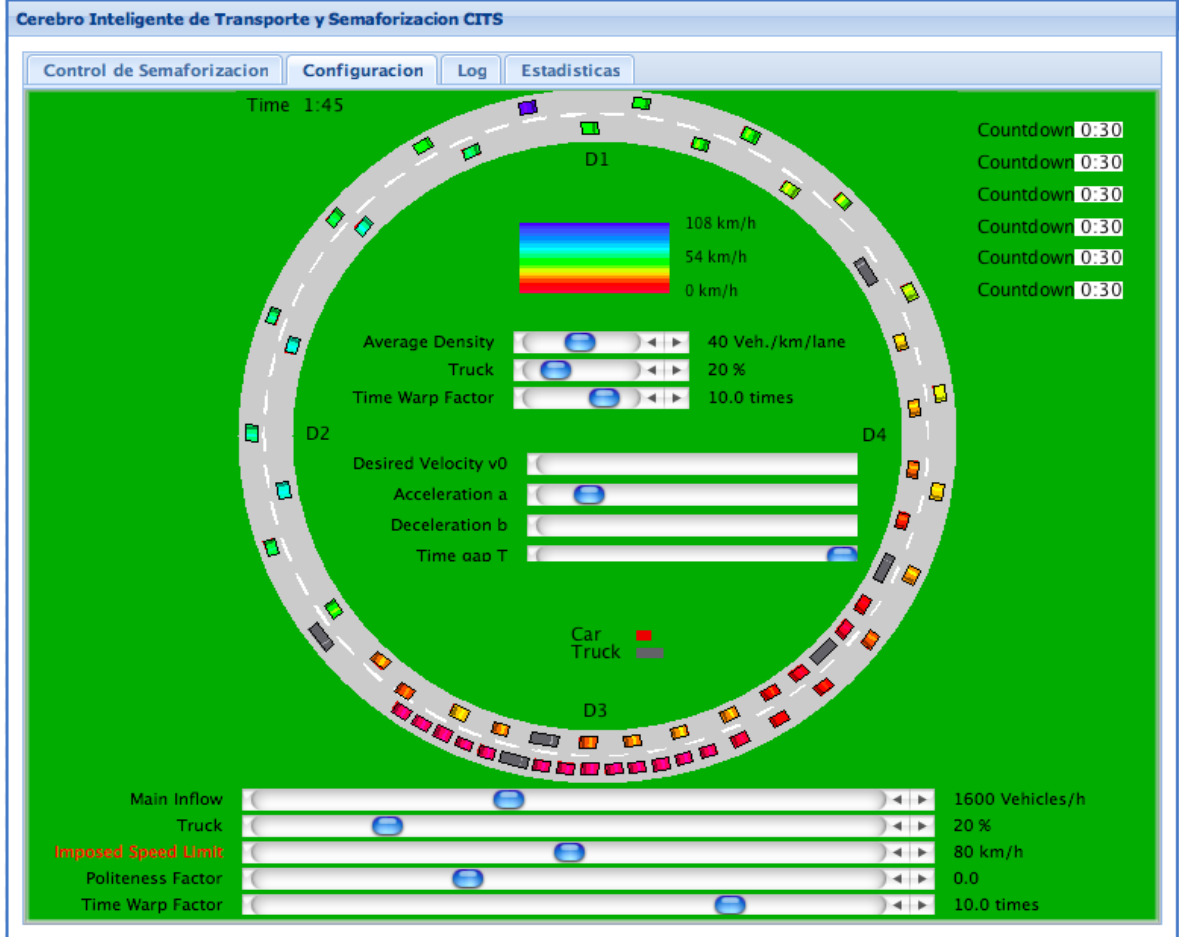


Imagen 7. Modulo 2 Configuracion Semaforica Actual

Fuente: Autor.

### 5.2.2.3 Modulo 3. Log de Cambios

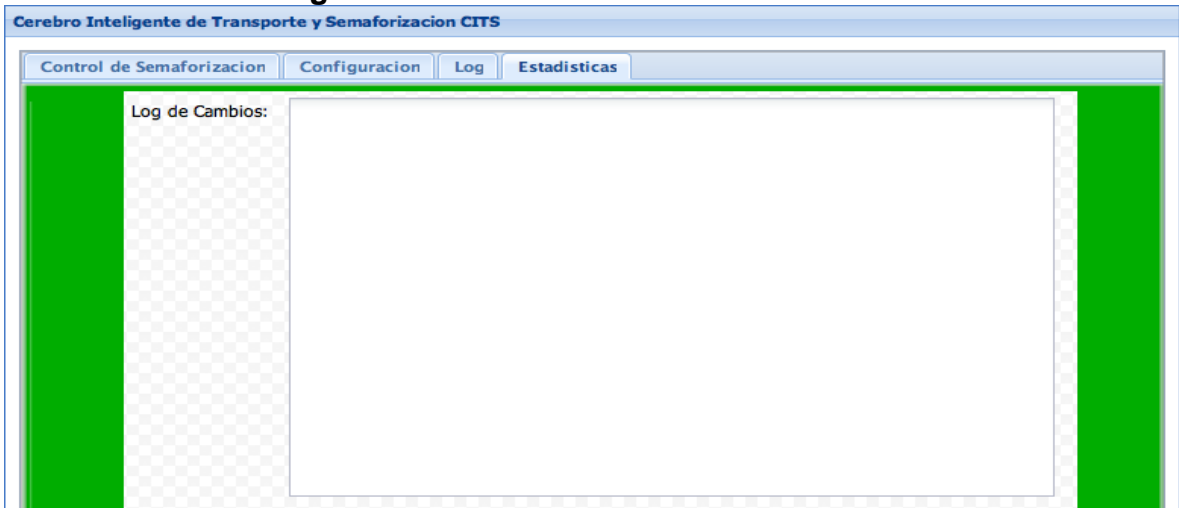


Imagen 8. Modulo 3 Log de Cambios

Fuente: Autor.

#### 5.2.2.4 Modulo 4. Estadísticas del Simulador

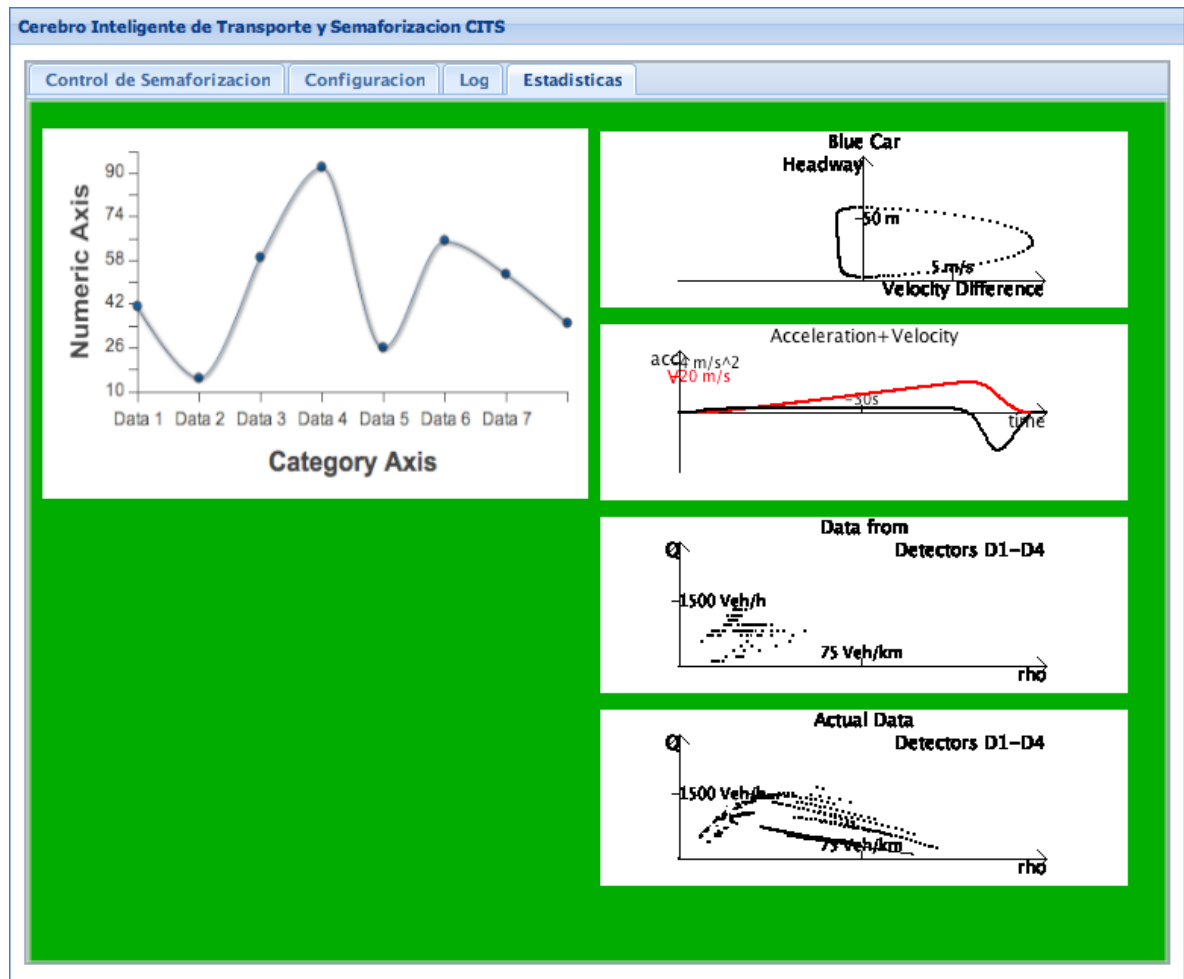


Imagen 9. Modulo 4 Estadísticas del Simulador

Fuente: Autor.

#### 5.2.3 Diagrama de alto nivel (Arquitectura)

Muestra la forma en que se encuentran conectados cada uno de los módulos que finalmente componen todo la solución.

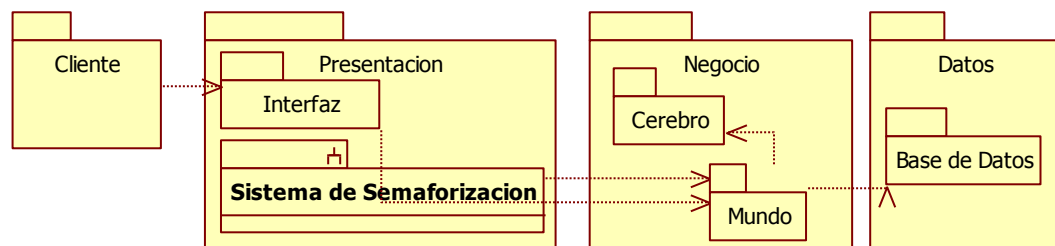


Imagen 13. Arquitectura

Fuente: Autor.

### 5.3 IMPLEMENTACION

Este proceso comenzó con una investigación sobre las tecnologías de simulación y los métodos inteligentes a trabajar en el proyecto.

En primer lugar jugué literalmente con un proyecto de trafico de simulación hecho en java, que describo en detalle en la sección de simuladores de trafico, al que modifique y cacharrea entre otras cosas, para saber hasta que punto podía manipular y entender el código que fue desarrollado por un alemán, lei algunos capítulos de un libro que enlazaba el autor del simulador en su pagina, un libro de simulación en el que el se había basado y así entendí varios conceptos de simulación que aplique en el método inteligente y pienso relatar en detalle al final del desarrollo del producto.

Luego conseguí el apoyo de un simulador de trafico mucho mas sofisticado con la diferencia que es pago y no open source, la empresa desarrolladora alemana del producto me ofreció su apoyo con la condición de entregar mi proyecto en pdf al finalizarlo. El contacto se realizó gracias al experto en sistemas de semaforización del SIT de America Klaus Banse quien accedió a ayudarme con el proyecto como asesor, luego de varios correos y entrevistas que tuve con el. Ya cuando se realizó el trámite de la licencia con fines educativos para uso en el proyecto me puse en el trabajo de entender el simulador y todas sus fantásticas herramientas con las que cuenta, este simulador es mucho mas sofisticado y aprenderlo fue un poco mas complejo que el simulador de trafico hecho en java, en parte por que no podía simplemente leer el código cuando no entendía alguna función, luego de entenderlo, se hicieron unas pruebas de conectividad con un hardware que de una caja de una intersección semafórica, el resultado fue optimo, la imagen de la caja puede visualizarse en la imagen 14.

Se comenzó a desarrollar el modulo 1 control de semaforización inteligente a moldear y a interactuar con el método inteligente que controlara los tiempos de estos, esta fase aun esta en desarrollo y por el tiempo los otros modulos aun están planeados sin desarrollarse.

Siguiendo con el marco de los métodos inteligentes se utilizó en el proyecto algoritmos genéticos y se comenzó su desarrollo, aunque antes de empezar evalúe cuales opciones resultaban optimas para la implementación y podían tomar una decisión racional en el proyecto, también hice algunos desarrollos para saber como se utilizaban.



Imagen 14. VTI Conexión de semáforos del SIT de America

Fuente: Klaus Banse.

Luego de desarrollar el primer modulo 1 control de semaforización inteligente, se desarrolla el modulo 2 configuracion semaforico actual ya que esta directamente ligado al primer modulo, y luego el modulo 3 log de cambios y se finaliza con el modulo 4 estadisticas del simulador ya que se desea probar este modulo con el resto de modulos finalizados, sin embargo cada modulo será puesto a prueba como un componenete a parte en las pruebas JUnit.

### 5.3.1 Metodos Inteligentes

#### 5.3.1.1 Modelo de tráfico longitudinal: El IDM <sup>1</sup>

En esta simulación, se utilizó el modelo inteligente-Driver (IDM) para simular la dinámica longitudinal, es decir, aceleraciones y desaceleraciones de frenado de los conductores.

El IDM es un "auto-siguiendo el modelo", es decir, el estado del tráfico en un momento dado se caracteriza por las posiciones, velocidades, y el índice de carril de



los vehículos. La decisión de cualquier conductor para acelerar o frenar depende sólo de su propia velocidad, y en el "vehículo de delante" inmediatamente delante de él. Las decisiones de cambio de carril, sin embargo, depende de todos los vehículos vecinos. En concreto, la aceleración  $dv / dt$  de un conductor determinado depende de su velocidad  $v$ ,  $s$  en la distancia con el vehículo delantero, y de la velocidad  $v$  Delta diferencia (positiva al acercarse),

$$\frac{dv}{dt} = a \left[ 1 - \left( \frac{v}{v_0} \right)^\delta - \left( \frac{s^*}{s} \right)^2 \right]$$

Imagen 14. Ecuacion IDM

Fuente: Autor.

donde

$$s^* = s_0 + \min \left[ 0, \left( vT + \frac{v\Delta v}{2\sqrt{ab}} \right) \right]$$

Imagen 15. Ecuacion IDM

Fuente: Autor.

La aceleración se divide en una "deseada" aceleración  $a [1 - (v / v_0)^\delta]$  en el buen camino libre, y reducir la velocidad de frenado inducida por el vehículo del frente. La aceleración en el buen camino libre disminuye a partir de la aceleración inicial de un a cero al acercarse a la "velocidad deseada"  $v_0$ . El término de frenado se basa en una comparación entre la "distancia deseada dinámica"  $s^*$ , y la  $s$  real diferencia con el vehículo precedente. Si la brecha real es aproximadamente igual a la  $s^*$ , entonces la desaceleración romper esencialmente compensa la parte de aceleración en vacío, así que la aceleración resultante es casi cero. Esto significa,  $s^*$  corresponde a la brecha al seguir a otros vehículos en constante flujo de tráfico. Además,  $s^*$  aumenta dinámicamente cuando se aproxime a los vehículos más lentos y disminuye cuando el vehículo delantero es más rápido. Como consecuencia de ello, la desaceleración del impuesto se incrementa con la

- disminución de la distancia al vehículo delantero (se quiere mantener una cierta "distancia de seguridad")
- incremento de la velocidad propia (que aumenta la distancia de seguridad)
- aumento de la diferencia de velocidad en el vehículo delantero (cuando se aproxima el vehículo frente a una tasa muy alta, una situación peligrosa puede ocurrir).

## Los parámetros del modelo

El IDM tiene parámetros intuitiva:

- deseado la velocidad cuando se conduce por una carretera gratuita,  $v_0$
- desea avanzar tiempo de seguridad al seguir a otros vehículos,  $T$
- la aceleración en el tráfico cotidiano,  $a$
- "Cómoda" la desaceleración de frenado en el tráfico cotidiano,  $b$
- mínimo de parachoques a parachoques distancia con el vehículo delantero,  $s_0$
- aceleración exponencial,  $\delta$ .

En general, todos los "conductores de vehículos de unidad" puede tener su propio conjunto de parámetros individuales, por ejemplo,

- camiones se caracterizan por los bajos valores de  $v_0$ ,  $a$ ,  $b$ , y,
- controladores de unidad de cuidado en un  $T$  avanzado de alta seguridad de tiempo,
- agresivos ("insistente") de los conductores se caracterizan por una  $T$  de baja en relación con los altos valores de  $v_0$ ,  $A$  y  $B$ .

A menudo dos tipos son suficientes para mostrar los principales fenómenos. Los parámetros estándar que se utiliza en las simulaciones son los siguientes:

Parámetro	Vehículos de	Valor de camiones	Comentarios
Deseada velocidad $v_0$	120 kmh	80 kmh	Para el tráfico de la ciudad, uno adaptar la velocidad deseada, mientras que el resto de parámetros se puede esencialmente sin cambios.
Tiempo avanzar $T$	1,5 s	1,7 s	Recomendación en las escuelas de conducción alemán: 1,8 s; valores realistas variar entre 2 s y 0,8 s, e incluso por debajo.
Distancia mínima $s_0$	2,0 m	2,0 m	Mantenerse en reposo completo, también en las colas que son causadas por un semáforo en rojo.
La aceleración de $a$	$0,3 \text{ m/s}^2$	$0,3 \text{ m/s}^2$	Valores muy bajos para mejorar la formación de parada de tráfico, y se van. Los valores reales son de $1-2 \text{ m/s}^2$
Desaceleración $b$	$3,0 \text{ m/s}^2$	$2,0 \text{ m/s}^2$	Valores muy altos para mejorar la formación de parada de tráfico, y se van. Los valores reales son de $1-2 \text{ m/s}^2$

Tabla 44. Parametros estadar IDM

Fuente: Autor.

### 5.3.1.2 Algoritmos Geneticos Modelo Inteligente

Se plantea la situación del modelo inteligente de manejo, y ahora se necesita que un modelo inteligente sea capaz de tomar las decisiones adecuadas para controlar los tiempos de los semáforos, siguiendo la premisa de que el método de algoritmos genéticos y aunque este no siempre dara una decisión adecuada siempre llegara a una solución razonablemente buena.

Los Algoritmos Geneticos<sup>2</sup> son metodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimizacion. Estan basados en el proceso genetico de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza de acorde con los principios de la seleccion natural y la supervivencia de los mas fuertes, postulados por Darwin (1859). Por imitacion de este proceso, los Algoritmos Geneticos son capaces de ir creando soluciones paraproblemas del mundo real. La evolucion de dichas soluciones hacia valores optimos del problemadepende en buena medida de una adecuada codicacion de las mismas.

Los principios basicos de los Algoritmos Geneticos fueron establecidos por Holland (1975), y se encuentran bien descritos en varios textos { Goldberg (1989), Davis (1991), Michalewicz (1992),Reeves (1993) } .

En la naturaleza los individuos de una poblacion compiten entre si en la busqueda de recursos tales como comida, agua y refugio. Incluso los miembros de una misma especie compiten a menudo en la busqueda de un companero. Aquellos individuos que tienen mas exito en sobrevivir y en atraer companeros tienen mayor probabilidad de generar un gran numero de descendientes. Por el contrario individuos poco dotados produzcan un menor numero de descendientes. Esto significa que los genes de los individuos mejor adaptados se propagaran en sucesivas generaciones hacia un numero de individuos creciente. La combinacion de buenas características provenientes de diferentes ancestros, puede a veces producir descendientes "superindividuos", cuya adaptacion es mucho mayor que la de cualquiera de sus ancestros. De esta manera, las especies evolucionan logrando unas características cada vez mejor adaptadas al entorno en el que viven.

Los Algoritmos Geneticos usan una analogia directa con el comportamiento natural. Trabajan con una poblacion de individuos, cada uno de los cuales representa una solucion factible a un problema dado. A cada individuo se le asigna un valor o puntuacion, relacionado con la bondad de dicha solucion. En la naturaleza esto equivaldria al grado de efectividad de un organismo para competir por unos determinados recursos. Cuanto mayor sea la adaptacion de un individuo al problema, mayor sera la probabilidad de que el mismo sea seleccionado para reproducirse, cruzando su material genetico con otro individuo seleccionado de igual forma. Este cruce producir a nuevos individuos descendientes de los anteriores los cuales comparten algunas de las características de sus padres. Cuanto menor sea la adaptacion de un individuo, menor sera la probabilidad de que dicho individuo sea

seleccionado para la reproducción, y por tanto de que su material genético se propague en sucesivas generaciones.

Planteando la introducción de cómo funcionan los algoritmos genéticos podemos describir el funcionamiento dentro del algoritmo inteligente que controlara los tiempos de los semáforos, siguiendo el esquema del diagrama de clases que muestra los pasos del caso de uso 003 cu-003 control inteligente de semaforización, el ambiente en el que se desarrolla es un simulador de tráfico, y esto hace que el algoritmo cree nuevas decisiones y pueda madurar a partir de tomar estas decisiones evaluando las que tenga un menor impacto en el sistema y menor tiempo.

De esta manera se produce una nueva población de posibles soluciones, la cual reemplaza a la anterior y verifica la interesante propiedad de que contiene una mayor proporción de buenas decisiones en comparación con las decisiones anteriores. A lo largo de las generaciones las buenas características se propagan a través de la población. Favoreciendo el cruce de los individuos mejor adaptados, van siendo exploradas las áreas más prometedoras del espacio de búsqueda. Si el Algoritmo Genético se diseña de la forma adecuada, la población convergerá hacia una solución óptima del problema.

---

2. Algoritmos Genéticos, . <http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/temageneticos.pdf>

### **5.3.2 Herramientas Estadísticas**

Para la creación de vehículos aleatoriamente, se deben tener en cuenta factores como la hora, que estimula la cantidad de pasajeros, y la cantidad de vehículos, todo esto con el fin de crear un ambiente parecido al real, y poder probar la efectividad de un ente inteligente que controle los semáforos para disminuir la densidad del tráfico.

También se utilizan herramientas estadísticas en las proyecciones de una densidad de automóviles y articulados en el futuro, esto se plantea con métodos estocásticos y regresión lineal, un proceso que está en fase de desarrollo con el asesor de estadística ya que se desea implementar cuando todos los componentes estén terminados, es decir el módulo de estadísticas será el último en desarrollarse como se planteó en la fase de implementación.

#### **5.3.2.1 Distribución Normal**

Esta es una distribución de Gauss, es una distribución de probabilidad de variable continua que con frecuencia aparece en fenómenos reales.

La gráfica de su función de densidad tiene una forma acampanada y es simétrica respecto de un determinado parámetro. Esta curva se conoce como campana de Gauss.

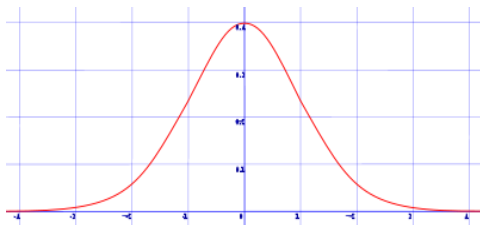


Imagen 12. Campana de Gauus

Fuen-

te: <http://es.wikipedia.org/wiki/Archivo:DisNormal01.svg>.

La importancia de esta distribución radica en que permite modelizar numerosos fenómenos naturales, sociales y psicológicos. Mientras que los mecanismos que subyacen a gran parte de este tipo de fenómenos son desconocidos, por la ingente cantidad de variables incontrolables que en ellos intervienen, el uso del modelo normal puede justificarse asumiendo que cada observación se obtiene como la suma de unas pocas causas independientes.

### 5.3.2.2 Distribución de weibull.

La distribución de Weibull complementa a la distribución exponencial y a la normal, se usa cuando se sabe de antemano que una de ellas es la que mejor describe la distribución de fallos o cuando se han producido muchos fallos (al menos 10) y los tiempos correspondientes no se ajustan a una distribución más simple.

La distribución de Weibull nos permite estudiar cuál es la distribución de fallos de un componente clave de seguridad que pretendemos controlar y que a través de nuestro registro de fallos observamos que éstos varían a lo largo del tiempo y dentro de lo que se considera tiempo normal de uso.

### 5.3.2.3 Distribución de poisson.

En teoría de probabilidad y estadística, la distribución de Poisson es una distribución de probabilidad discreta. Expresa la probabilidad de un número  $k$  de eventos ocurriendo en un tiempo fijo si estos eventos ocurren con una tasa media conocida, y son independientes del tiempo desde el último evento.

La distribución fue descubierta por Siméon-Denis Poisson (1781–1840) que publicó, junto con su teoría de probabilidad, en 1838 en su trabajo *Recherches sur la probabilité des jugements en matières criminelles et matière civile* ("Investigación sobre la probabilidad de los juicios en materias criminales y civiles"). El trabajo estaba enfocado en ciertas variables aleatorias  $N$  que cuentan, entre otras cosas, un número de ocurrencias discretas (muchas veces llamadas "arribos") que tienen lugar durante un intervalo de tiempo de duración determinada. Si el número esperado de ocurrencias en este intervalo es  $\lambda$ , entonces la probabilidad de que haya exactamente  $k$  ocurrencias (siendo  $k$  un entero no negativo,  $k = 0, 1, 2, \dots$ )

## 5.3.3 Simuladores de Trafico

### 5.3.3.1 VISSIM - Multi-Modal Traffic Flow Modeling

VISSIM es el principal programa de simulación microscópica de multi-modal modelado de flujo de tráfico. Con su único nivel alto de detalle que simula con precisión el tráfico urbano y de carretera, incluidos los ciclistas y vehículos motorizados.



Imagen 14. Simulacion Vissim Fuente: Autor.

VISSIM es la herramienta ideal para profesionales del transporte que quieren simular diferentes escenarios de tráfico antes de iniciar la aplicación. Por lo tanto, les permite encontrar una solución que tenga la calidad de tráfico y transporte, seguridad y costo en consideración. Como VISSIM combina la experiencia de ingeniería de tráfico y el estado de la técnica de las opciones de presentación, animaciones 3D, incluso, no sólo es usado por los profesionales del transporte. Cada vez más los responsables políticos y las autoridades locales están eligiendo VISSIM demostrar de manera convincente la eficacia de una medida proyectada podría ser, independientemente de si una nueva carretera que va a ser construido o una nueva línea de tranvía se está planeando. VISSIM por lo tanto, ofrece la oportunidad única de integrar a los ciudadanos en el proceso de toma de decisiones.

Implementar el algoritmo inteligente en VISSIM ahorra tiempo valioso al proyecto en el desarrollo o la creación de un simulador de trafico, y se utilizan herramientas muy buenas que trae este paquete, se pueden obtener resultados y mostrarlos de manera eficiente y sencilla en este simulador, para conseguir la licencia de ayuda fue necesario explicar el proyecto a los creadores del software que están muy interesados en el desarrollo del proyecto, el contacto se pudo realizar gracias al experto de sistemas inteligentes de transporte Klaus Banse de SIT de America.

Fuente: <http://www.vissim.de/index.php?id=1801>

Video: <http://www.vissim.de/software/transportation-planning-traffic-engineering/software-system-solutions/vissim/example-of-use/>

### 5.3.3.2 Simulador de trafico en Java

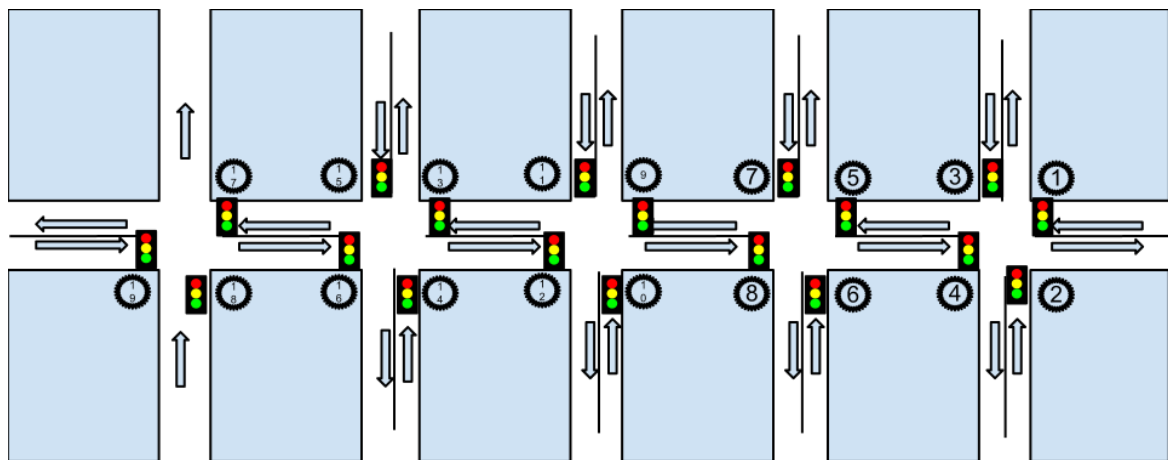


Imagen 15. Simulador Java

Fuente: Autor.

Este es un applet hecho en Java por Martin Treiber, un experto en Alemania sobre simulación de tráfico, creo este modelo de tráfico en Java y libero su código con una licencia Open Source del MIT de EUA, he modificado el código y hecho algunos ajustes así que implementare tanto en VISSIM como en este simulador en Java.

Fuente: <http://vwisb7.vkw.tu-dresden.de/~treiber/MicroApplet/>

### 5.3.4 Documentación del Código

Para una comprensión rápida del código por entes externos al proyecto o por el mismo autor en el momento del mantenimiento, todo el código implementado debe ser documentado de la siguiente manera:

Documentación en Java

- Todas las clases tienen la definición de su uso en forma general
- Todos los métodos y/o funciones se definen con sus parámetros de entrada y salida con ejemplo de estos en caso de contar con ellos
- Todos los comentarios son expresados con “//”
- Las funciones y/o métodos expresadas dentro de otro proceso se deben especificar en base a su funcionalidad de forma general
- En caso de excepciones indicar la razón de esta y especificar sus caminos consiguientes

### 5.3.5 Distribución de Actividades de Codificación

Dada a la complejidad del proyecto y como se expresa en el tema de Dificultades Técnicas donde se encontraron problemas con los simuladores de tráfico y otros problemas en el transcurso del proyecto dados en el aprendizaje del software con licencia de proyecto de grado dado por PTV Traffic Customerservice por esto y el

tiempo resulto insuficiente en varias ocasiones por cuestiones ajenas al proyecto por lo que los tiempos tuvieron que ser ampliados por causas de fuerza mayor.

Tabla 44. Distribución de actividades y cronograma, 2011 2012 Fuente: Autor.

### **5.3.6 Mantenimiento**

El mantenimiento en este proyecto de software está dado por las condiciones de mejora que se encuentren en la interacción del proyecto con el usuario final, en la construcción esto está dado por la interacción de los hitos con los asesores del proyecto y otros entes involucrados en la participación como expertos en movilidad y asesores del sistema de transporte Transmilenio, también por el mismo desarrollador que en este caso es el autor del proyecto a través de las pruebas.

Es importante tanto la documentación detallada invertida en este proyecto como la documentación del código, para que este proyecto pueda ser continuado con fines educativos en algunas de las materias de la carrera o para optimizar el algoritmo inteligente mejorando los objetivos del proyecto, también como uso del autor para en caso de que se den las condiciones para que sea implementado en la vida real, o algún ente externo quiera observar el desarrollo del proyecto de forma detallada.

## **5.4 PRUEBAS**

Se realizan 3 clases de pruebas:

- Pruebas del sistema
- Pruebas unitarias
- Pruebas de integración

Estas pruebas están en detalle en el anexo de pruebas ya que se encuentra en desarrollo.

## **6. SUPUESTOS Y DIFICULTADES**

### **6.1 ASESORIA**

En algunos factores importantes del proyecto fue importante conseguir algún experto que la universidad no contaba, como un experto en movilidad, inteligencia artificial, sistemas inteligentes y estadística. Esta búsqueda estuvo marcada por la disponibilidad de los asesores para atender mis solicitudes y mis preguntas también fue importante el primer contacto con personas a las que no conocía, como por ejemplo el Señor Klaus Banse, experto en sistemas inteligentes de transporte del SIT de América, que con su apoyo obtuve una licencia de VISSIM de la em-



presa alemana PTV Traffic Customerservice, el cual viaja constante por su trabajo y el tiempo de asesoría y apoyo era muy distante.

## **6.2 DIFICULTADES AL OBTENER LA INFORMACIÓN**

La información del sistema de semaforización utilizado en la ciudad de Bogota D.C. actualmente que esta enunciado en el marco teorico, fue dado por la secretaria de movilidad de manera muy formal y amable, mi solicitud fue atendida en un tiempo minimo, las dificultades se presentaron al pedir información técnica a los ingenieros de siemens que operaban los equipos de siemens que controlan los semáforos y el programa con el que se cambian actualmente, ya que es propiedad de siemens y el protocolo de comunicación también, esto como dicen los ingenieros de siemens es importante por la seguridad, pero no con la interoperabilidad ya que ningún otro software que no se de siemens puede interactuar con este, también su sistema es uno de los mas antiguos del mercado, aun en la época en la que fue adquirido por el distrito y aunque funciona actualmente ya no cumple con las necesidades de una ciudad de constante crecimiento.

## **6.3 DIFICULTADES TECNICAS**

Conseguir un simulador de trafico para evitar consumir tiempo en el diseño y la realización de este fue unos de los primeras dificultades, mientras se tramitaba la licencia de VISSIM que cuenta con un retroalimentación de los mapas de Google Maps, con funciones de generación de variables y otros métodos y funciones adicionales, encontré un simulador de trafico hecho en Java, que contaba con lo necesario, aunque no con las especificaciones estadísticas que yo había planteado algo que se desarrollo en este simulador y el mapa del alcance también especificado que se cambio mientras se interactuo con este simulador.

Al conseguir el apoyo y la licencia de VISSIM interactuar con este es un poco mas complejo, con el simulador de JAVA yo podía modificar lo que deseara pero con VISSIM había que leer los manuales de usuario en Aleman y entender su lógica, aunque aprender como manejar VISSIM fue mas de la interaccion de este al nivel del conocido “cacharreo” luego al conseguir los manuales en ingles pude saber lo que hacia en mayor detalle.

## **6.4 RESULTADOS**

Al finalizar el desarrollo de este proyecto, se han concretado los siguientes resultados:

- Construcción de una herramienta de un prototipo de un sistema inteligente de semaforización con las características de un proyecto de divulgación académica de los algoritmos genéticos altamente usable para quienes quieren entender el tema y estudian o cursan ingeniería de sistemas.

- Aplicación de las etapas del ciclo en vida de software en un desarrollo iterativo para dar solución a una necesidad de tráfico en la ciudad con un algoritmo inteligente capaz de controlar los semáforos y los ciclos del tiempo del mismo.
- Aplicación de la arquitectura de la tecnología Java para lograr software de calidad escalable, mantenible.
- La fase de desarrollo de la aplicación presentó una desviación de tiempos de aproximadamente un 48% en relación con lo planeado en el cronograma inicial.
- Se realizó la creación de las primeras generaciones a partir de la demanda, luego el algoritmo genético adaptaba estos valores y los cruzaba y mutaba, para aprender de los ciclos de los tiempos de los semáforos.
- Se evidencia que controlando los ciclos de los tiempos de los semáforos en tiempo real y dependiendo de la demanda vehicular además de utilizar los algoritmos genéticos para la optimización y el aprendizaje se puede minimizar los tiempos en que un articulado (bus del Transmilenio) va desde el punto de inicio de la simulación al punto final.
- Utilizando el simulador movsim y aplicando la teoría de algoritmos genéticos se creó un escenario para la experimentación del tráfico con inteligencia artificial, para probar y estudiar los métodos de inteligencia artificial.

**Ver ANEXO H.**

## **7. CONCLUSIONES Y TRABAJO FUTURO**

### **7.1 CONCLUSIONES**

Este prototipo presenta una alternativa para la mejora de la movilidad, aunque para ello se requiere de una cantidad de información real importante para que los datos simulados se aproximen más a la realidad.

Es importante destacar que gracias al simulador de tráfico movsim, fue posible realizar el experimento, el cual se utilizó para probar el algoritmo genético con un tráfico simulado, en el marco científico el software libre contribuye a la expansión del conocimiento y esto ayudó para poder comprobar las teorías y objetivos planteados en el inicio de este proyecto.

Los resultados del algoritmo genético aunque en un principio no parecieron prometedores el experimento demostró que mejoraría continuamente el tráfico.

Los resultados de un 17% de disminución de la densidad vehicular, una mejora en la velocidad promedio de un 38.1 % y en la velocidad máxima promedio de un 7.8% aproximadamente, que conllevaría a mejorar la movilidad tanto para los articulados del sistema Transmilenio y para los automóviles particulares, también es importante destacar que estos porcentajes podrían ser mucho mayores por la cantidad de iteraciones del algoritmo genético, entre más veces sea iterado el proceso de negocio el resultado será mucho más cercano a un resultado óptimo.

El equipo utilizado en las pruebas del experimento tuvo problemas de rendimiento, por ello es necesario contar con un buen equipo para realizar mejores pruebas que exigen más recursos de hardware.

Las pruebas realizadas requieren de mucho más tiempo para tomar buenos resultados respecto a algoritmo genético.

Aunque la cantidad de vehículos del sistema Transmilenio en las horas de pico parece baja, es importante destacar que se tienen que tener factores en cuenta para cumplir la demanda, ya que tiene que mantener una distancia de seguridad entre articulados para evitar acciones de choque, por eso la cantidad de articulados no puede superar un tope aun en las horas pico.

Los conocimientos adquiridos durante la carrera proporciona la manera adecuada de abordar los problemas a los que es enfrentado un ingeniero de sistemas.

Las etapas de análisis y diseño son importantes mucho más que la implementación ya que proporcionan las bases adecuadas para realizar un buen proyecto de software y al momento de llegar a la implementación son mucho menos preguntas las que el desarrollador tiene que hacer sobre cómo abordar el problema.

Las buenas prácticas de desarrollo facilitan la implementación del código y lo hacen mucho más rápido de igual manera al momento de llegar al mantenimiento del software se hace más sencillo de abordar.

La implementación estadística en un proyecto de este alcance facilita la muestra de resultados y da la confianza que estos resultados estén cercanos a la realidad.

## **7.1 TRABAJO FUTURO**

Este trabajo puede ser evaluado para ser implementado en la ciudad de Bogotá, según los experimentos realizados la implementación mejoraría la movilidad en un porcentaje óptimo aunque estos datos pueden cambiar a favor al aumentar las horas de prueba.

En la implementación real, pueden ser creadas futuras investigaciones de como el sensor captura la información y como hacer esta información aún más valiosa, capturando mayor cantidad de detalles, por el alcance del proyecto el sensor solo devuelve la cantidad de vehículos, pero sin duda un sensor real podrá capturar la velocidad de los vehículos y utilizar esta información para alimentar el algoritmo genético.

Se podría con la información captada realizar unas proyecciones de la densidad del tráfico, el modelo quedaría completo si se toman los procesos estocásticos para realizar esta tarea, pero el sólo comprender el tema tomaría entre uno o dos semestres más, así que es bueno tener en cuenta este ítem como la continuación

de un proyecto, o como una opción de trabajo para la maestría y porque no para un doctorado.

También puede ser mejorado el algoritmo genético por ejemplo en la investigación se pudo observar que el uso de un operador de cruce uniforme ha demostrado ser más eficiente en otros contextos que el cruce en un punto, si se probara de esta manera podría determinar en que medida mejora cambiar el punto de cruce.

Aunque en esta investigación se utiliza algoritmos genéticos por ser un método de optimización y por ser un método evolutivo que se adapta particularmente a la idea de mejoramiento continuo al sistema de semaforización, no quiere decir que sea el único método inteligente a utilizar para la optimización de un sistema de semaforización, por ello esta investigación puede ser tomada como punto de partida para implementar otros métodos inteligentes.

## **7.2 IMPLEMENTACION EN LA VIDA REAL**

Al tratar de solucionar los problemas de tráfico de una ciudad, en general hablamos de implementar un modelo de optimización con un conjunto de parámetros y variables, como la densidad vehicular, la distancia entre semáforos, los tiempos de duración de los ciclos de los semáforos y un objetivo específico, como la cantidad de paradas de los vehículos, la cantidad de gasolina gastada, el tiempo total de espera en el semáforo, en este caso y por el alcance dado inicialmente en el proyecto se intenta mejorar el tiempo en que los articulados del sistema Transmilenio van de un punto a otro, y minimizar el tiempo de espera en el semáforo. El problema en si es bastante complejo ya que interfieren bastantes variables en el sistema, por ejemplo si solo se da tiempo por demanda, a mayor demanda en un semáforo más tiempo en el ciclo verde tendría y eso causaría congestión en las vías contiguas con lo cual podría ocurrir un colapso en el sistema, aunque la implementación del algoritmo genético determina entonces cuales son los más indicado para ser colocados así eliminando esta situación, otro ejemplo el ciclo y la desfase son variables dependientes y esto hace que al modificar una se tenga que modificar la otra, por ejemplo el sistema tiene que tener en cuenta que el tiempo rojo en un semáforo, debe contar como el verde en el contiguo, es decir que además de los vehículos en cola en el propio semáforo debe estar atento de los semáforos contiguos en la intersección.

## **7.3 RECOMENDACIONES**

De acuerdo a la experiencia recogida en este proyecto y conociendo las necesidades del usuario se sugiere:

- Si se desea implementar otro método inteligente, es necesario leer y entender el objetivo y la manera es que se aborde con algoritmos genéticos, para que pueda enfocar el objetivo e implementar el método inteligente.
- Tener una formación estadística e interés en la inteligencia artificial y la simulación.
- Se recomienda a futuro diseñar el algoritmo genético paralelo.

- A partir de la experiencia recogida aquí el paso a una implementación de varios métodos inteligentes sería esencial para determinar cual puede ser el óptimo en la implementación de un sistema semafórico.

## 8. BIBLIOGRAFÍA

1. Ignacio Hernández Molina, *LA INVESTIGACION CIENTIFICA Un camino a la imaginación*, Bogotá D.C., 2002
2. Azarang, Mohammad R. - García Dunna, Eduardo, *Simulación y análisis de modelos estocásticos*, MCGRAW HILL, 1996
3. Secretaria de Movilidad de Bogotá, *SISTEMA DE CONTROL SEMAFÓRICO DE BOGOTÁ D.C*, 2011
4. Congested Traffic States in Empirical Observations and Microscopic Simulations, Martin Treiber, Ansgar Hennecke, Dirk Helbing. Autor URL: <http://www.mtreiber.de/>, <http://xxx.uni-augsburg.de/abs/cond-mat/0002177>, 2000  
30 Agosto
5. Verkehrsdynamik und -simulation, Martin Treiber, Arne Kesting <http://141.30.182.245/verkehrsdynamik/index.shtml>, 2010
6. Jong K.A., Spears W.M., y Gordon D.F.. Using genetic algorithms for concept learning, (1993)
7. Azarang, R. y García, E. Simulación y análisis de modelos estocásticos, Ed. McGraw Hill 1a ed. (1996)
8. Laurence A., Adapting Operator Probabilities in Genetic Algorithms, (1995)
9. Spears W. y Anand V. A Study Of Crossover Operators In Genetic Programming , 1991.
10. Iba E., Análisis y Diseño de Algoritmos Genéticos Paralelos Distribuidos, 1999
11. Treiber M., Hennecke A. and Helbing D. Congested Traffic States in Empirical Observations and Microscopic Simulations, (2000)
12. IDM Informacion: <http://www.vwi.tudresden.de/~treiber/MicroApplet/IDM.html>
13. Thrun S. y Norvig P., Artificial Intelligence, Stanford Engineering, <https://www.ai-class.com/> 2011.

14. Germ R., Kesting A., BuddenM. y Treiber M. Información sobre el simulador, <http://www.h1451310.stratoserver.net/movsimwebpage/index.shtml>, 2012.
15. J.H. Holland. Adaptation in natural and artificial systems. University ofMichigan Press, 1975. Reimpreso por MIT Press en 1992.
16. Guerra Hernandez A. Aprendizaje Automático: Algoritmos genéticos. 2004.
17. IBM, The Globalization of Traffic Congestion, 2010.
18. Baeck, T., Hoffineister, F., & Schwefel, H. A survey of evolution strategies. Proceedings of the Fourth International Conference on Genetic Algorithms (pp. 2-9). La Jolla, CA: Morgan Kaufmann. 1991.
19. DiCyt, Información sobre el trafico y el sistema de semaforización de la ciudad de Bogota, <http://www.dicyt.com/noticias/bogota-estudia-la-implantacion-de-sistemas-inteligentes-de-regulacion-del-traffic>. 2008.
20. Banse K., Protocolos Its Para Sistemas De Semaforización, <http://www.sitcolombia.com/download/presentaciones/201109%2029%20co%20pso%20avante%20protocolos.pdf>, 2011.
21. Ignacio Hernández Molina, *LA FORMULACION DE PROYECTOS En Ciencias e Ingenierias*, Bogotá D.C., 2012
22. García R., Utilización de Algoritmos Genéticos en Sistemas Inteligentes Autóomos, <http://iidia.com.ar/rgm/tesistas/steinhilber-tesis-de-grado-en-sistemas-de-informacion.pdf>, 2009.



ANEXOS

Anexo A. Ambiente del Algoritmo Genetico

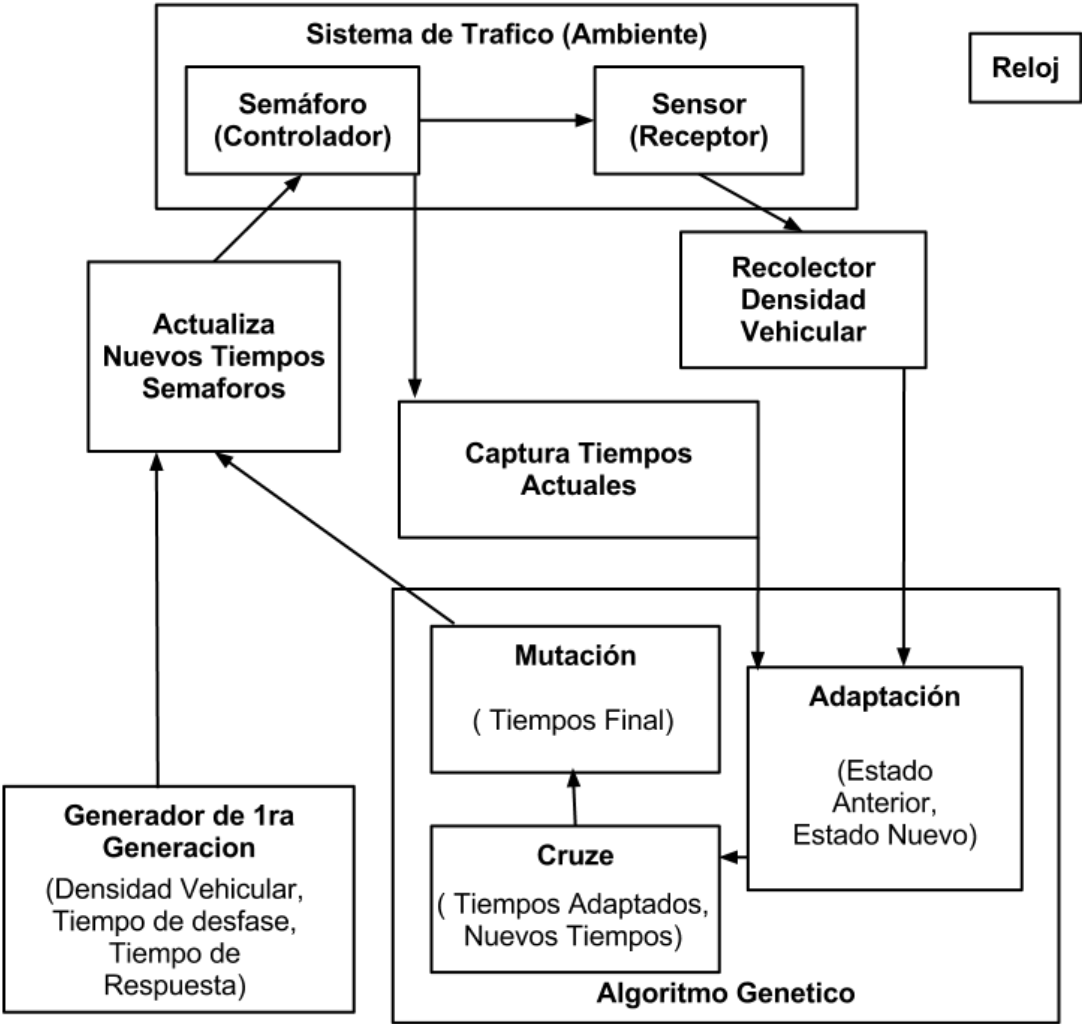


Imagen 16. Ambiente Algoritmo genetico

Fuente: Autor.

Anexo B. Cruce y Mutacion Algoritmos Geneticos

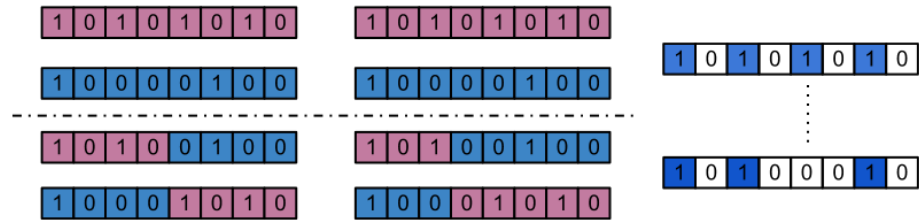


Imagen 17. Cruce y Mutacion

Fuente: Autor.

Anexo C. Interfaz de usuario



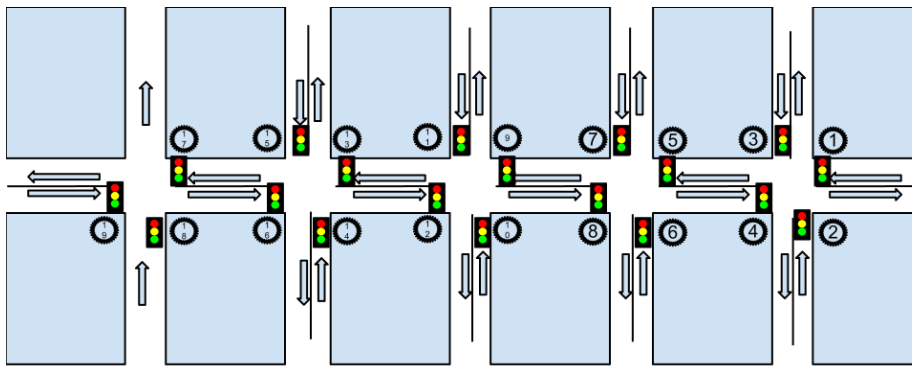


Imagen 18. Interfaz de usuario esquema inicial

Fuente: Autor.

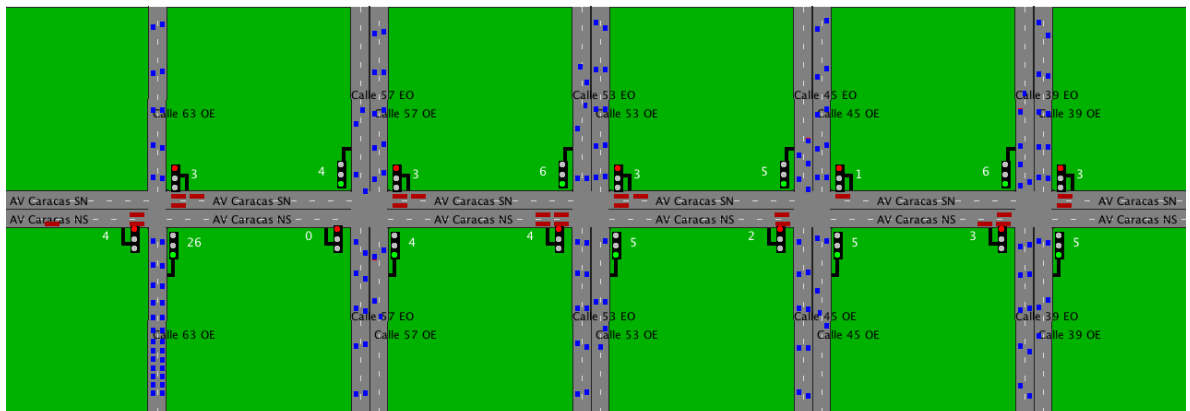


Imagen 19. Interfaz de usuario final

Fuente: Autor.

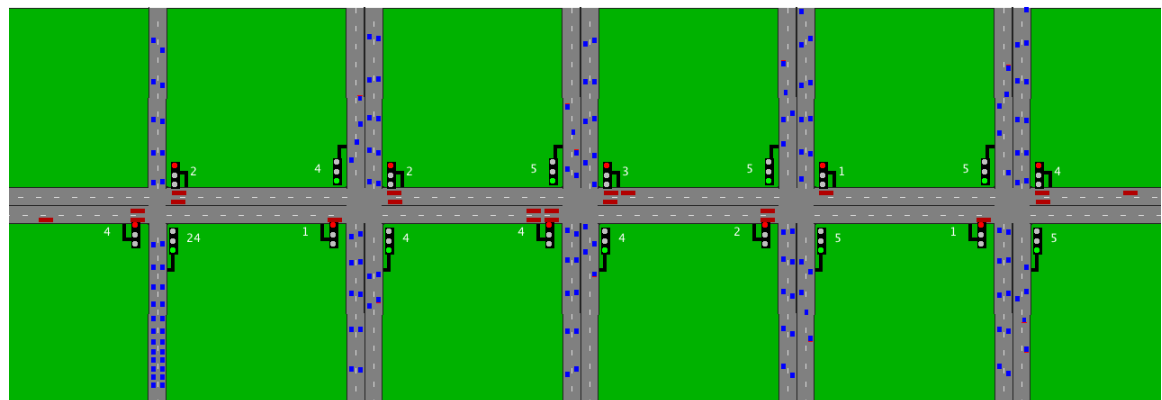
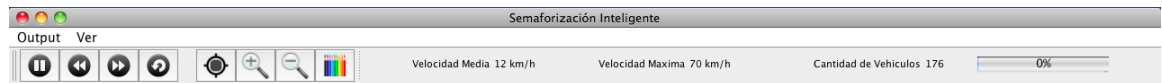


Imagen 20. Interfaz de usuario final

Fuente: Autor.

## Anexo D. Pruebas Unitarias

Estas pruebas corresponden a las pruebas de funcionalidad de los modulos.

PRECONDICIONES DE CASO DE PRUEBA			
Empezar Proyecto, Instanciar variables, importar clases, crear simulacion			
CASOS DE PRUEBA			
Clase	Método	C. Prueba	Resultado
Cerebro	main	Ejecución	Satisfactorio
PROCESOS DE PRUEBAS			
OBJETIVOS		CRITERIOS	
Crear una instancia de la clase recibe que recibe los parámetros base de un A.C.		Cada uno de los atributos internos de la clase deben tener los valores asignados.	
PROCESO		RESPONSABLE	
<ul style="list-style-type: none"><li>Se definen los valores de prueba, para el algoritmo genético.</li><li>Se instancia la clase y se hace la generación y se muestra su resultado en consola.</li></ul>		Franz Joseph Rogelez Carvajal	
PRUEBA			
La instancia se creó sin problemas y en consola apareció la instancia creada esperada.			

PRECONDICIONES DE CASO DE PRUEBA			
Crear Mapa			
CASOS DE PRUEBA			
Clase	Método	C. Prueba	Resultado
Mapa	crearMapa	Creacion	Satisfactorio
PROCESOS DE PRUEBAS			
OBJETIVOS		CRITERIOS	
Crear el mapa de las calles y los semáforos, en el mundo		Ejecutar el método de prueba de modulo	
PROCESO		RESPONSABLE	
<ul style="list-style-type: none"><li>Se crea una instancia de la clase</li><li>Se llama al método crearMapa</li></ul>		Franz Joseph Rogelez Carvajal	
PRUEBA			
La instanciación se realizo satisfactoriamente. Los valores que se generaron corresponden al mapa los semáforos. Instancia clase semáforo.			

PRECONDICIONES DE CASO DE PRUEBA			
Inicializar tiempos en semáforos.			
CASOS DE PRUEBA			
Clase	Método	C. Prueba	Resultado
Cerebro	asignarTiemposNulos	Generación	Satisfactorio
PROCESOS DE PRUEBAS			
OBJETIVOS		CRITERIOS	
Recorre todo el mapa y instancia los semáforos con tiempos en cero.		Ejecutar el método de prueba de modulo	
PROCESO		RESPONSABLE	
<ul style="list-style-type: none"><li>Se crea una instancia de la clase</li><li>Se llama al método asignarTiemposNulos</li></ul>		Franz Joseph Rogelez Carvajal	
PRUEBA			
La instanciación se realizo satisfactoriamente. Los valores que se generaron corresponden a los tiempos iniciales.			

PRECONDICIONES DE CASO DE PRUEBA			
Actualizar Tiempos de los semáforos.			
CASOS DE PRUEBA			
Clase	Método	C. Prueba	Resultado
Cerebro	asignarTiempos	Generación	Satisfactorio
PROCESOS DE PRUEBAS			
OBJETIVOS		CRITERIOS	
Asignar tiempos a los semaforos		Ejecutar el método de prueba de modulo	
PROCESO		RESPONSABLE	
<ul style="list-style-type: none"><li>Se crea una instancia de la clase</li><li>Se llama al método asignarTiempos</li></ul>		Franz Joseph Rogelez Carvajal	
PRUEBA			
La instanciación se realizo satisfactoriamente. Los valores de prueba arrojaron un resultado satisfactorio, los tiempos de los semáforos corresponden a los generados por el sistema.			

PRECONDICIONES DE CASO DE PRUEBA			
Capturar Densidad			
CASOS DE PRUEBA			
Clase	Método	C. Prueba	Resultado
Cerebro	capturaDensidad	Generación	Satisfactorio
PROCESOS DE PRUEBAS			
OBJETIVOS		CRITERIOS	
Capturar la densidad generada por la simulacion		Ejecutar el método de prueba de modulo	
PROCESO		RESPONSABLE	
<ul style="list-style-type: none"><li>Se crea una instancia de la clase</li><li>Se llama al método capturaDensidad</li></ul>		Franz Joseph Rogelez Carvajal	
PRUEBA			
Capturar la cantidad de vehículos en cada semáforo que ha generado el sistema			
PRECONDICIONES DE CASO DE PRUEBA			
Simulador haber generador vehículos en el sistema de trafico			
CASOS DE PRUEBA			
Clase	Método	C. Prueba	Resultado
Sensor	densidad	Generación	Satisfactorio
PROCESOS DE PRUEBAS			
OBJETIVOS		CRITERIOS	
Captura Densidad		Ejecutar el método de prueba de modulo	
PROCESO		RESPONSABLE	
<ul style="list-style-type: none"><li>Se crea una instancia de la clase</li><li>Se llama al método densidad</li></ul>		Franz Joseph Rogelez Carvajal	
PRUEBA			
Se recorren los semáforos y a través del sensor se captura la densidad generada.			

## Anexo E. Pruebas del Sistema

Identificador de la prueba	PS – 1
Estado de la prueba	Aceptado
Tipo de prueba	SISTEMA
Objetivo de la prueba	Evaluar el desempeño de la maquina virtual cuando se generan los tiempos, la densidad vehicular, la simulación, la creación de generaciones de tiempos en los algoritmos genéticos. Verificar tiempo de respuesta.
Descripción	Se evalúa el rendimiento de la aplicación para los algoritmos genéticos.
Entrada	Generaciones de algoritmos genéticos. Tiempo de respuesta.
EJECUTADO POR	
Franz Joseph Rogelez Carvajal	
VERIFICADO POR	
Franz Joseph Rogelez Carvajal	
RESULTADOS ESPERADOS	
A lo mucho 10 segundos.	
COMENTARIOS	
Se obtuvo un resultado entre 0.03 y 80 segundos después de 50 iteraciones.	
CRITERIO DE ACEPTACIÓN	
El tiempo de respuesta es acorde con las especificaciones de la maquina.	

Tiempo de respuesta.

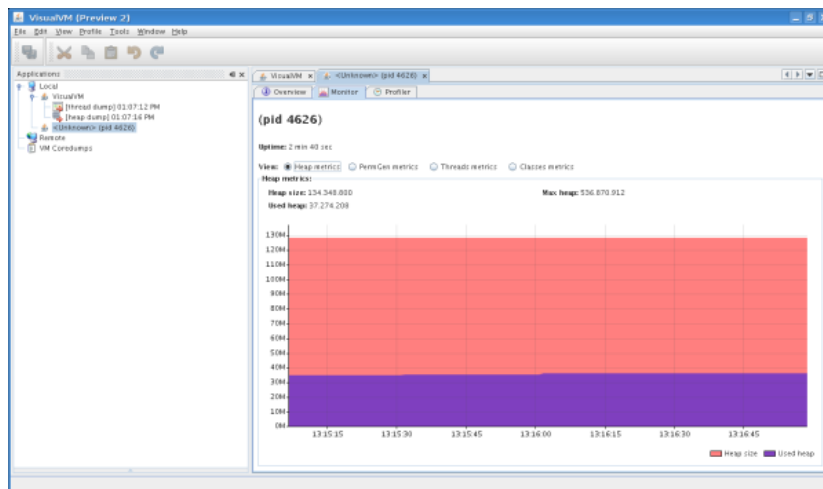
```

239643 [Thread-3] INFO (Simulator.java:timeStep:521) - Simulator.update :time = 20.00s = 0.01h, dt = 0.20s, projectName=trafficlight
242319 [Thread-3] INFO (Simulator.java:timeStep:521) - Simulator.update :time = 40.00s = 0.01h, dt = 0.20s, projectName=trafficlight
244988 [Thread-3] INFO (Simulator.java:timeStep:521) - Simulator.update :time = 60.00s = 0.02h, dt = 0.20s, projectName=trafficlight
247713 [Thread-3] INFO (Simulator.java:timeStep:521) - Simulator.update :time = 80.00s = 0.02h, dt = 0.20s, projectName=trafficlight

```

Fuente: El autor

Prueba de consumo de memoria en el uso del los algoritmos geneticos.



Fuente: El autor

## ANEXO F. PRUEBAS DE INTEGRACION

<b>Identificador de la prueba</b>	PI – 1
<b>Estado de la prueba</b>	Aceptado
<b>Tipo de prueba</b>	Ejecución de la aplicación
<b>Objetivo de la prueba</b>	Validar el inicio del simulador
<b>EJECUTADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>VERIFICADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>RESULTADOS ESPERADOS</b>	
Inicio satisfactorio del generador	
<b>COMENTARIOS</b>	
Inicio satisfactorio del generador	
<b>CRITERIO DE ACEPTACIÓN</b>	
Inicio satisfactorio del generador	

<b>Identificador de la prueba</b>	PI – 2
<b>Estado de la prueba</b>	Aceptado
<b>Tipo de prueba</b>	Generación de los tiempos de los ciclos de los semáforos
<b>Objetivo de la prueba</b>	Validar la generación de tiempos para los semáforos
<b>EJECUTADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>VERIFICADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>RESULTADOS ESPERADOS</b>	
Generación de primeras generaciones, y generaciones a partir del cruce y mutación	
<b>COMENTARIOS</b>	
Se crea los tiempos a partir de las demandas y luego se verifica en la función de adaptación	
<b>CRITERIO DE ACEPTACIÓN</b>	
Tiempos Generados	

<b>Identificador de la prueba</b>	PI – 3
<b>Estado de la prueba</b>	Aceptado
<b>Tipo de prueba</b>	Función de adaptación
<b>Objetivo de la prueba</b>	Verifica si los tiempos fueron o no adaptables al sistema de tráfico
<b>EJECUTADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>VERIFICADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>RESULTADOS ESPERADOS</b>	
Tiempos adaptados cumpliendo las condiciones esperadas.	
<b>COMENTARIOS</b>	
Se validan los tiempos para los semáforos	
<b>CRITERIO DE ACEPTACIÓN</b>	
Los tiempos que permanecen realmente son los óptimos	

<b>Identificador de la prueba</b>	PI – 4
<b>Estado de la prueba</b>	Aceptado
<b>Tipo de prueba</b>	Creacion del trafico
<b>Objetivo de la prueba</b>	Ver el trafico en el simulador
<b>EJECUTADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>VERIFICADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>RESULTADOS ESPERADOS</b>	
Visualizacion del trafico, y vista desde el mundo	
<b>COMENTARIOS</b>	
Se puede ver el trafico y el mundo lo capta de la misma manera	
<b>CRITERIO DE ACEPTACIÓN</b>	
Que sea visible para el usuario el trafico y el algoritmo genetico también lo pueda ver.	

<b>Identificador de la prueba</b>	PI – 5
<b>Estado de la prueba</b>	Aceptado
<b>Tipo de prueba</b>	Generacion de los ciclos cruzados y mutados
<b>Objetivo de la prueba</b>	Verificar que se mutan y cruzan los tiempos
<b>EJECUTADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>VERIFICADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>RESULTADOS ESPERADOS</b>	
Los tiempos cruzados y mutados son actualizados en la interfaz	
<b>COMENTARIOS</b>	
Es visible el cambio de rango de los ciclos de los semaforos	
<b>CRITERIO DE ACEPTACIÓN</b>	
Los tiempos deben ser diferentes a los tiempos creados	

<b>Identificador de la prueba</b>	PI – 6
<b>Estado de la prueba</b>	Aceptado
<b>Tipo de prueba</b>	Guarda el objeto Array de semáforos en la base de datos
<b>Objetivo de la prueba</b>	Persistir el trafico acumulado
<b>EJECUTADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>VERIFICADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>RESULTADOS ESPERADOS</b>	
Una tabla con el objeto guardado por el DAO	
<b>COMENTARIOS</b>	
Es visible el objeto	
<b>CRITERIO DE ACEPTACIÓN</b>	
Que sea guardado el objeto, y pueda se consultado.	

<b>Identificador de la prueba</b>	PI – 7
<b>Estado de la prueba</b>	Aceptado
<b>Tipo de prueba</b>	Estadística a partir de la información guardada.
<b>Objetivo de la prueba</b>	Estadística a partir de la información guardada.
<b>EJECUTADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>VERIFICADO POR</b>	
Franz Joseph Rogelez Carvajal	
<b>RESULTADOS ESPERADOS</b>	
Que sea visible la mejora del tráfico, con los cambios de los algoritmos genéticos	
<b>COMENTARIOS</b>	
Al variar los ciclos de los tiempos, se puede ver que la gráfica cambia la pendiente	
<b>CRITERIO DE ACEPTACIÓN</b>	
Que sea visible la información del resultado de los algoritmos genéticos.	

## ANEXO G. ESTADÍSTICAS PARA LOS DATOS INICIALES

### Transporte Urbano Área Metropolitana de Bogotá Transmilenio<sup>1</sup>

ESTADÍSTICAS TRANSMILENIO 2012		
DATOS GENERALES		
Pasajeros totales al 30 de abril de 2012	3.803.439.499	Pasajeros
Promedio pasajeros hora pico de abril de 2012	199.511	Pasajeros
Pasajeros alimentados totales al 30 de abril de 2012 (entradas/salidas)	1.897.639.441	Pasajeros
Pasajeros intermunicipales totales al 30 de abril de 2012	222.718.783	Pasajeros
Estaciones en operación	115	Estaciones
Kilómetros de vía en operación troncal	87	Km
Flota troncal disponible vinculados 30 de abril de 2012	1.291	Buses
Velocidad promedio flota troncal abril de 2012	26,08	Km/hora
Promedio kilómetros recorridos flota troncal abril de 2012	338.060	Km
Rutas alimentadoras	83	Rutas
Flota alimentación vinculados 30 de abril de 2012	526	Buses
barrios alimentados (aprox)	318	Barrios
Km en operación de alimentación (aprox)	663	Km

**Imagen 21. Transmilenio Estadísticas Fuente: [www.transmilenio.gov.co](http://www.transmilenio.gov.co)**

**[http://www.transmilenio.gov.co/WebSite/Contenido.aspx?ID=TransmilenioSA\\_TransmilenioEnCifras\\_EstadisticasGenerales](http://www.transmilenio.gov.co/WebSite/Contenido.aspx?ID=TransmilenioSA_TransmilenioEnCifras_EstadisticasGenerales)**

En el trimestre abril – junio de 2011, el servicio de transporte Transmilenio registró los siguientes resultados:

Un parque automotor de 1.269 buses troncales (promedio diario para el trimestre), de los cuales 1.240 buses se encontraban en servicio y en los que se transportaron 120,3 millones de pasajeros. Esta cifra fue equivalente a 29,1% del total de personas movilizadas en el área metropolitana de Bogotá (413,6 millones de pasajeros). De 519 buses alimentadores, se utilizaron en promedio 488, que transportaron 60,7 millones de pasajeros. Con respecto al segundo trimestre del año anterior, se presentó un aumento de 5,7% en el número de pasajeros transportados por el sistema (Cuadro 1 y Tabla 5).

Los buses troncales representaron 6,6% del parque automotor de la ciudad. Teniendo en cuenta todo el Sistema Transmilenio (incluyendo alimentadores) esta proporción asciende a 9,4% de los automotores de transporte urbano de pasajeros de Bogotá. El sistema contó con 114 estaciones para los buses troncales en 84 kilómetros de vías en operación. Los buses alimentadores trabajaron en 74 rutas, en un recorrido de 515 kilómetros de vías.



**Comportamiento por modos de transporte, en 8 áreas metropolitanas y 15 ciudades**  
**II trimestre (2010 - 2011)**

Tipo de vehículo	Variables	II Trim 2010	II Trim 2011 <sup>F</sup>	Variación % Anual	Variación % Año Corrido	Variación % 12 Meses
Total	Promedio mensual de vehículos afiliados	47.343	45.587	-3,7	-3,3	-3,0
	Promedio mensual de vehículos en servicio	39.902	38.703	-3,0	-2,8	-2,6
	Pasajeros transportados (miles)	969.106	962.039	-0,7	-0,5	-0,8
Buses	Promedio mensual de vehículos afiliados	13.812	13.093	-5,2	-5,3	-5,0
	Promedio mensual de vehículos en servicio	11.477	10.880	-5,2	-5,2	-4,6
	Pasajeros transportados (miles)	267.792	256.078	-4,4	-4,6	-5,0
Busetas	Promedio mensual de vehículos afiliados	13.454	12.769	-5,1	-5,0	-5,3
	Promedio mensual de vehículos en servicio	11.590	11.140	-3,9	-4,2	-4,8
	Pasajeros transportados (miles)	247.123	238.882	-3,3	-4,2	-5,3
Microbuses y colectivos	Promedio mensual de vehículos afiliados	17.552	16.947	-3,4	-2,8	-2,5
	Promedio mensual de vehículos en servicio	14.512	14.047	-3,2	-3,0	-2,7
	Pasajeros transportados (miles)	271.897	265.441	-2,4	-1,2	-1,7
Transmilenio (Bogotá)	Promedio mensual de vehículos afiliados*	1.692	1.788	5,7	6,5	8,4
	Promedio mensual de vehículos en servicio*	1.600	1.728	8,0	9,1	11,2
	Pasajeros transportados (miles)	113.869	120.334	5,7	5,8	6,1
Megabús (Pereira)	Promedio mensual de vehículos afiliados*	140	144	3,1	2,6	2,2
	Promedio mensual de vehículos en servicio*	131	133	1,8	0,1	0,1
	Pasajeros transportados (miles)*	8.766	8.601	-1,9	-1,5	-2,0
MIO (Cali)	Promedio mensual de vehículos afiliados**	443	465	5,0	13,1	35,2
	Promedio mensual de vehículos en servicio**	362	427	17,9	28,4	54,3
	Pasajeros transportados (miles)**	15.799	22.666	43,5	50,2	73,1
Metrolínea (Bucaramanga) <sup>1</sup>	Promedio mensual de vehículos afiliados**	124	132	6,7	--	--
	Promedio mensual de vehículos en servicio**	112	108	-3,3	--	--
	Pasajeros transportados (miles)**	5.234	4.480	-14,4	--	--
Metrolínea (Bucaramanga) <sup>1</sup>	Promedio mensual de vehículos afiliados**	124	132	6,7	--	--
	Promedio mensual de vehículos en servicio**	112	108	-3,3	--	--
	Pasajeros transportados (miles)**	5.234	4.480	-14,4	--	--
Cable (Manizales) <sup>2</sup>	Promedio mensual de vehículos afiliados	42	42	-	-2,4	--
	Promedio mensual de vehículos en servicio	42	31	-26,2	-25,0	--
	Pasajeros transportados (miles)	429	414	-3,5	13,3	--
Transmetro (Barranquilla) <sup>3</sup>	Promedio mensual de vehículos afiliados**	--	123	--	--	--
	Promedio mensual de vehículos en servicio**	--	123	--	--	--
	Pasajeros transportados (miles)**	--	4.089	--	--	--
Metro (Medellín)	Promedio mensual de vehículos afiliados	126	126	-	--	--
	Promedio mensual de vehículos en servicio	118	117	-0,8	-1,7	-2,1
	Pasajeros transportados (miles)	38.198	41.056	7,5	7,3	7,8

Fuente: DANE - ETUP

- No hubo variación debido a que el parque automotor se mantiene igual.

\* Incluye buses troncales y alimentadores.

\*\* Incluye buses troncales, padrones y alimentadores.

<sup>1</sup> Metrolínea inició operaciones en el segundo trimestre de 2010, por lo tanto no se presentan variaciones año corrido ni 12 meses.

<sup>2</sup> Cable inició operaciones en el primer trimestre de 2010, por lo tanto no se presentan variaciones 12 meses.

<sup>3</sup> Este tipo de servicio inició sus operaciones en agosto de 2010, por lo tanto no se presentan datos para el segundo trimestre de 2010, variaciones anuales, año corrido y 12 meses.

<sup>F</sup> Cifra preliminar.

Imagen 22. Comportamiento Transporte Fuente: Dane [..\\Fuentes\\Estadisticas\\bolet\\_transp\\_IItrim11.pdf](#)

**Distribución del parque automotor y de pasajeros en el área metropolitana de Bogotá  
II trimestre 2011**

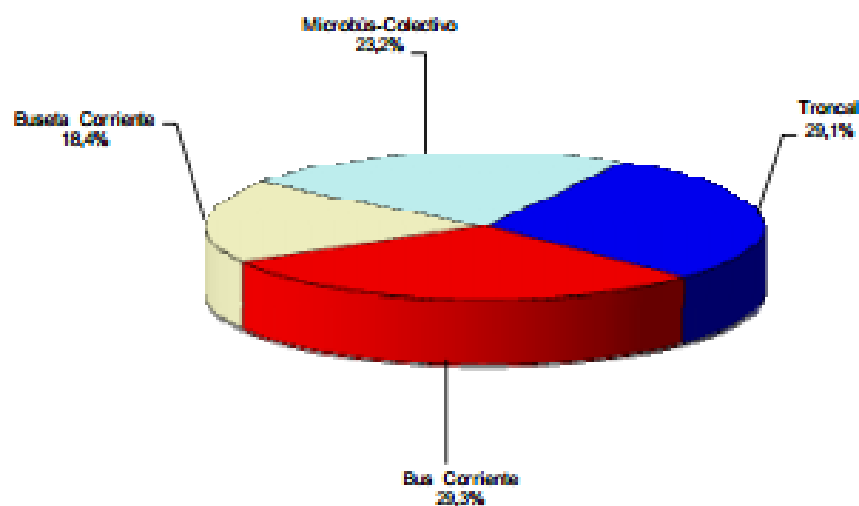
Tipo de vehículo	Promedio mensual de vehículos afiliados	% Vehículos	Total pasajeros transportados (miles)*	% Pasajeros
Área Metropolitana de Bogotá	19.081	100,0	413.563	100,0
Alimentadores	519	2,7	60.698	
Troncal	1.269	6,6	120.334	29,1
Bus Corriente	6.503	34,1	121.160	29,3
Buseta Corriente	4.343	22,8	76.006	18,4
Microbús-Colectivo	6.447	33,8	96.064	23,2

Fuente: DANE - ETUP

\* Se incluyen pasajeros transportados en buses alimentadores pero no se suman en el total, porque son registrados globalmente por Transmilenio.

Imagen 23. Distribucion Transmilenio Fuente: Dane [..\Fuentes\Estadisticas\bolet\\_transp\\_IItrim11.pdf](..\Fuentes\Estadisticas\bolet_transp_IItrim11.pdf)

**Distribución porcentual de los pasajeros transportados en el área metropolitana de Bogotá  
II trimestre 2011**



Fuente: DANE - ETUP

Imagen 24. Distribucion Transmilenio Fuente: Dane [..\Fuentes\Estadisticas\bolet\\_transp\\_IItrim11.pdf](..\Fuentes\Estadisticas\bolet_transp_IItrim11.pdf)

1. Departamento Administrativo Nacional de Estadística - DANE, Estadísticas Generales de Transporte Colombia, [http://www.dane.gov.co/files/investigaciones/boletines/transporte/bolet\\_transp\\_IItrim11.pdf](http://www.dane.gov.co/files/investigaciones/boletines/transporte/bolet_transp_IItrim11.pdf), Bogotá, D.C. - Colombia – Agosto de 2011

\*Los municipios que componen esta área metropolitana son: Bogotá, Cajicá, Cota, Chía, Funza, Gachancipá, Madrid, Mosquera, Sibate, Soacha, Tabio, Tenjo, Tocancipá, Zipaquirá.



## INFORMACIÓN ESTADÍSTICA

### Movimiento del parque urbano automotor y pasajeros transportados, según áreas metropolitanas y ciudades\* II trimestre (2010 - 2011)

Áreas Metropolitanas y Ciudades	Promedio mensual de vehículos afiliados			Promedio diario de vehículos en servicio			Total pasajeros transportados <sup>1</sup> Miles		
	2.010	2011 <sup>P</sup>	Variación %	2.010	2011 <sup>P</sup>	Variación %	2.010	2011 <sup>P</sup>	Variación %
Total general	47.343	45.587	-3,7	39.902	38.703	-3,0	969.106	962.039	-0,7
Área Metropolitana de Barranquilla	3.914	3.423	-12,5	3.503	3.138	-10,4	81.261	76.936	-5,3
Área Metropolitana de Bogotá	19.422	19.081	-1,8	15.986	15.739	-1,5	417.206	413.563	-0,9
Área Metropolitana de Bucaramanga	1.590	1.533	-3,6	1.528	1.467	-4,0	33.840	32.174	-4,9
Área Metropolitana de Cali	4.139	3.612	-12,7	3.238	2.966	-8,4	77.770	79.489	2,2
Área Metropolitana de Cúcuta	2.331	2.278	-2,3	1.862	1.810	-2,8	26.826	26.753	-0,3
Área Metropolitana de Manizales	933	939	0,7	682	766	12,2	17.280	18.293	5,9
Área Metropolitana de Medellín	5.422	5.261	-3,0	4.912	4.774	-2,8	139.862	144.637	3,4
Área Metropolitana de Pereira	769	771	0,3	713	711	-0,2	23.912	23.770	-0,6
Armenia	360	357	-0,7	337	343	1,9	4.776	4.427	-7,3
Cartagena	1.874	1.859	-0,8	1.574	1.557	-1,0	39.281	38.121	-3,0
Florencia	152	149	-1,5	122	111	-8,5	1.804	1.552	-14,0
Ibagué	1.195	1.182	-1,1	1.116	1.062	-4,9	20.876	18.136	-13,1
Montería	183	75	-59,1	162	74	-54,2	3.425	1.348	-60,6
Neiva	681	672	-1,3	550	551	0,1	6.812	6.711	-1,5
Pasto	503	498	-0,9	466	461	-1,2	8.237	9.186	11,5
Popayán	660	649	-1,7	581	574	-1,3	9.741	9.600	-1,4
Quibdó	190	198	4,2	119	91	-23,6	1.554	1.167	-24,9
Riohacha	72	74	2,3	29	27	-7,0	617	530	-14,1
Santa Marta	885	905	2,3	721	742	2,8	28.393	29.692	4,6
Sincedejo	177	176	-0,8	146	146	0,5	2.468	2.376	-3,7
Tunja	521	525	0,8	472	485	2,7	6.441	6.122	-5,0
Valledupar	361	354	-1,9	154	152	-0,9	2.228	2.151	-3,4
Villavicencio	1.008	1.015	0,6	931	957	2,8	14.497	15.303	5,6

FUENTE: DANE

<sup>1</sup> No incluye pasajeros transportados en buses alimentadores, porque son registrados globalmente por Transmilenio.

\* Incluye Transmilenio, Megabús, Mo, Cable, Metrolínea, Transmetro y Metro.

<sup>P</sup> Cifras preliminares

Imagen 22. Estadística Movimiento Transporte Fuente:Dane [..\\Fuentes\\Estadisticas\\anexos transp lItrim11.xls](#).



## INFORMACIÓN ESTADÍSTICA

### Movimiento de SITM, Cable y Metro y pasajeros transportados, según áreas metropolitanas y ciudades y nivel de servicio II trimestre (2010 - 2011)

Áreas Metropolitanas y Ciudades	Promedio mensual de vehículos afiliados			Promedio diario de vehículos en servicio			Total pasajeros transportados Miles <sup>1</sup>		
	2.010	2011 <sup>P</sup>	Variación %	2.010	2011 <sup>P</sup>	Variación %	2.010	2011 <sup>P</sup>	Variación %
Total SITM, Metro y Cable	2.401	2.565	6,8	2.211	2.437	10,2	182.294	201.638	10,6
Área Metropolitana de Bogotá	1.692	1.788	5,7	1.600	1.728	8,0	113.869	120.334	5,7
Alimentadores	516	519	0,6	486	488	0,4			
Troncal	1.176	1.269	7,9	1.114	1.240	11,3	113.869	120.334	5,7
Área Metropolitana de Cali	443	465	5,0	362	427	17,9	15.799	22.666	43,5
Alimentadores	120	128	6,6	102	118	15,3	2.194	3.177	44,8
Padrón	221	236	6,8	178	215	21,0	3.647	6.225	70,7
Troncal	102	101	-1,0	82	94	14,6	9.959	13.263	33,2
Área Metropolitana de Medellín	126	126	-	118	117	-0,8	38.198	41.056	7,5
Metro	126	126	-	118	117	-0,8	38.198	41.056	7,5
Área Metropolitana de Manizales	42	42	-	42	31	-26,2	429	414	-3,5
Cable	42	42	-	42	31	-26,2	429	414	-3,5
Área Metropolitana de Pereira	140	144	3,1	131	133	1,8	8.766	8.601	-1,9
Alimentadores	89	91	2,6	86	87	1,6	3.450	3.548	2,8
Troncal	51	53	3,9	45	46	2,2	5.315	5.053	-4,9
Área Metropolitana de Bucaramanga	124	132	6,7	112	108	-3,3	5.234	4.480	-14,4
Alimentadores	63	69	8,9	55	55	-0,6	1.229	927	-24,6
Padrón	45	47	3,7	42	40	-4,0	1.433	1.381	-3,6
Troncal	15	16	6,7	15	13	-11,1	2.572	2.172	-15,5
Área Metropolitana de Barranquilla	0	123	***	0	123	***	0	4.089	***
Alimentadores	0	73	***	0	73	***	0	1.323	***
Troncal	0	50	***	0	50	***	0	2.766	***

FUENTE: DANE

<sup>1</sup> Incluye pasajeros transportados en buses alimentadores para Transmilenio en Bogotá

\*\*\* Cálculo matemático indeterminado

<sup>P</sup> Cifras preliminares

Imagen 23. Estadística Movimiento Transmilenio Fuente:Dane [..\\Fuentes\\Estadisticas\\anexos transp lItrim11.xls](#).

## ANEXO H. RESULTADOS DEL EXPERIMENTO

Para comenzar este experimento se tomo un conjunto de datos reales extraídos de las estadísticas del tráfico de la ciudad de Bogotá, los cuales se convierten en los datos de entrada para crear el tráfico simulado, con esto es posible disponer de unos resultados más cercanos a la realidad.

El tráfico que se crea a partir de los datos encontrados en las estadísticas de la ciudad, son el conjunto de valores iniciales que determina la generación de números pseudoaleatorios que corresponden a la densidad vehicular generada para el prototipo. Esta generación de números pseudoaleatorios son creados utilizando un algoritmo creado con el estadista Dagoberto Saboyá, con de la información recolectada que es utilizada como semilla para generar el nuevo tráfico.

Con el tráfico creado a partir de los datos recolectados se obtiene una muestra que es una buena aproximación al tráfico real de la ciudad de Bogotá, que es utilizada para los dos escenarios de prueba y comparación.

En el primer escenario los tiempos de los ciclos (Rojo y Verde) de los semáforos son los mismos que tienen los semáforos en la ciudad de Bogotá y que fueron recogidos a través de una investigación que se realizó para saber el funcionamiento del sistema de semaforización real de la ciudad de Bogotá, esta investigación se encuentra en detalle en el documento completo de este proyecto.

En el segundo escenario los tiempos de los ciclos (Rojo y Verde) de los semáforos son los tiempos que el algoritmo genético asigna a cada semáforo.

Estos dos escenarios de prueba son comparados para visualizar en qué medida la implementación de algoritmos genéticos en el sistema de semaforización mejora la movilidad respecto al funcionamiento del sistema actual.

Graficas del total de transmlenios en la troncal a partir de información inicial, Ver anexo F.

Kilómetros de via en operación troncal: **87 Km**

Flota troncal disponible vinculados 30 de abril de 2012: **1.291 Buses**

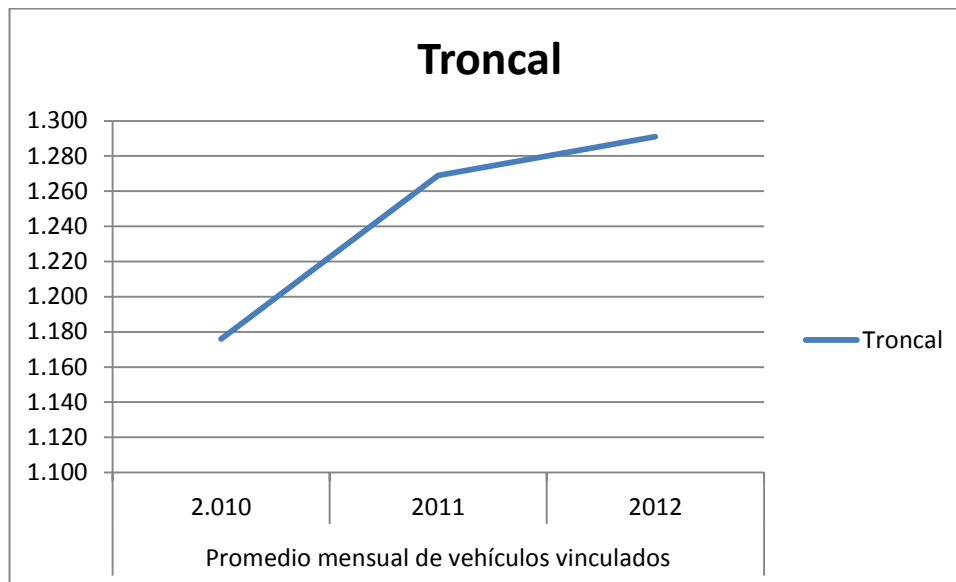
Velocidad promedio flota troncal abril de 2012: **26,08 Km/hora**

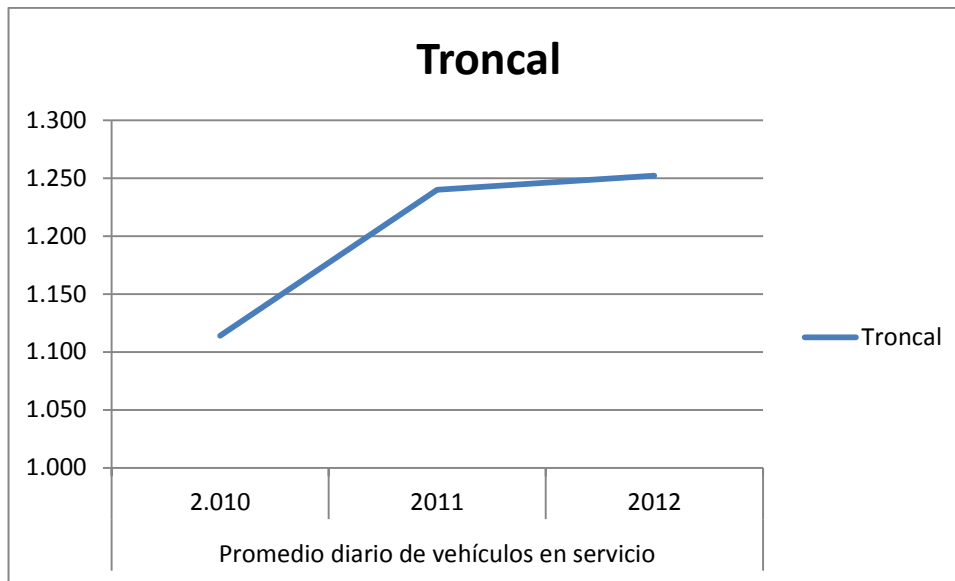
Promedio kilómetros recorridos flota troncal abril de 2012: **338.060 Km**

Áreas Metropolitanas y Ciudades	Promedio mensual de vehículos afiliados			Promedio diario de vehículos en servicio		
	2.010	2011 <sup>p</sup>	Variación %	2.010	2011 <sup>p</sup>	Variación %
Área Metropolitana de Bogotá	1.692	1.788	5,7	1.600	1.728	8,0
Troncal	1.176	1.269	7,9	1.114	1.240	11,3

## Vehículos en la troncal 2010 – 2012

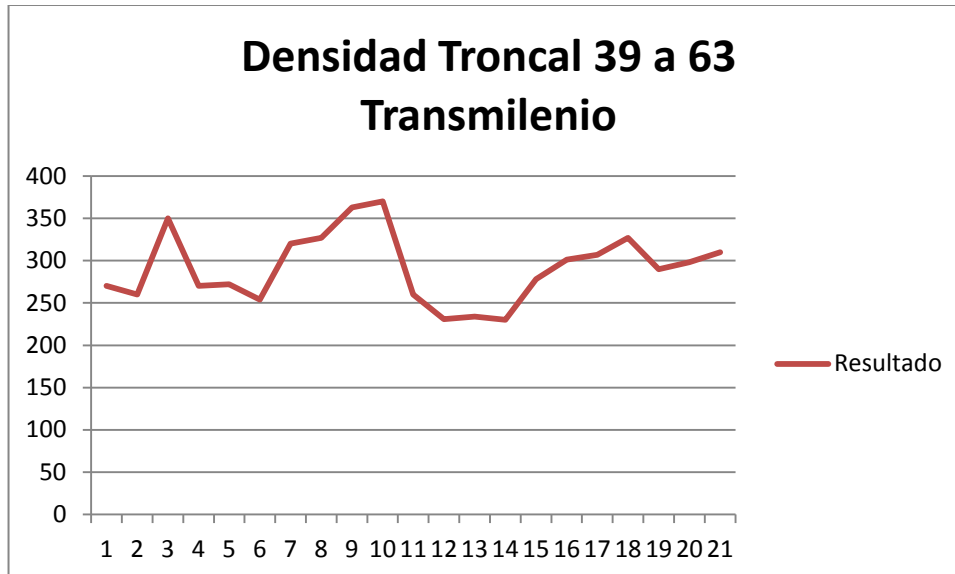
	Promedio mensual de vehículos vinculados			Promedio diario de vehículos en servicio		
	2.010	2011	2012	2.010	2011	2012
Troncal	1.176	1.269	1.291	1.114	1.240	1.252





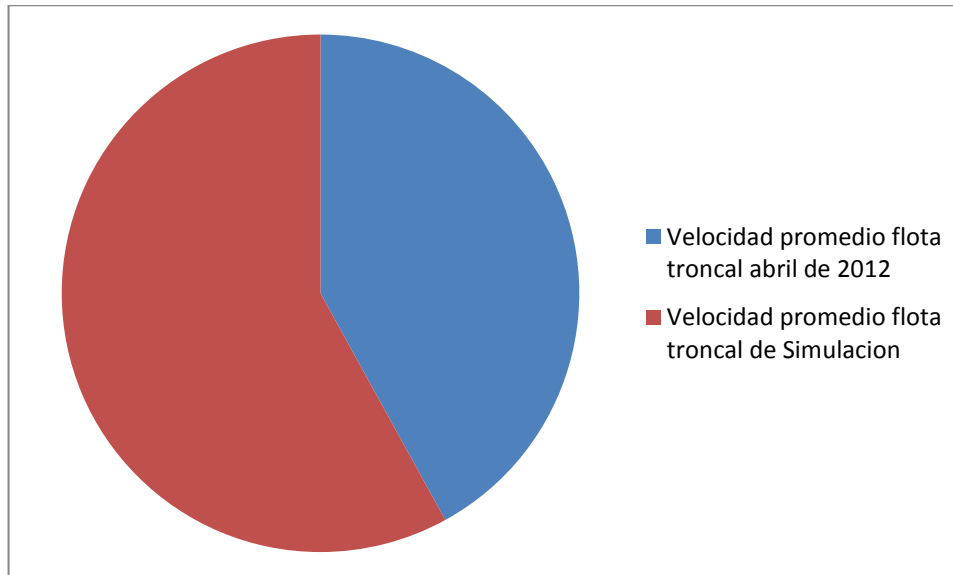
Datos Generados a partir de los datos recogidos.

En 21 días a partir del modelamiento estadístico de las variables se puede ver el resultado siguiente.



Movilidad a partir de los tiempos del sistema de semaforización actual.

Velocidad	V	Medida
Velocidad promedio flota troncal abril de 2012	26,08	km/h
Velocidad promedio flota troncal de Simulacion	36,03	km/h
Velocidad maxima promedio flota troncal Simulacion	68,06	km/h



Densidad Vehicular Simulada a partir de los datos reales, comparación entre los tiempos reales de los semáforos y los tiempos que genera el algoritmo genetico. Tomada la captura en un punto promedio.

Semaforo	Densidad Simulacion	Densidad Algoritmos Geneticos
1	10	6
2	15	16
3	31	31
4	15	10
5	13	12
6	15	10
7	13	12
8	10	6
9	9	5
10	6	3
11	5	3
12	10	6
13	15	16

<b>14</b>	13	15
<b>15</b>	10	8
<b>16</b>	40	32
<b>17</b>	13	10
<b>18</b>	12	8
<b>19</b>	6	6

